

LAB-10 : DB PROJECT

ASSIGNMENT 8-DDL & SQL

202318006

202318046

1. Create DDL Scripts

Address Table

AddressID	PK
Street	VARCHAR (10)
City	VARCHAR(10)
State	VARCHAR(10)
Zipcode	INT

Client Table

Client ID	PK
Name	VARCHAR NOT NULL
Budget	INT NOT NULL
Rating	INT
Contact	INT
AddressID	FK

Suppliers Table

Suppliers ID	PK
Name	VARCHAR (20) NOT NULL
Contact	INT NOT NULL
Portfolio Link	VARCHAR (15)
Experience	INT NOT NULL
Rating	INT

Designer Table

Designer ID	PK
Name	VARCHAR(20) NOT NULL
Contact	INT NOT NULL
Portfolio Link	VARCHAR (10)
Experience	INT NOT NULL

Project Table

Project ID	PK
Designer ID	FK
Client ID	FK
Start Date	DATE NOT NULL
End Date	DATE NOT NULL
Status	VARCHAR (10) NOT NULL
Cost	INT NOT NULL

Appointment Table

Appointment ID	PK
Designer ID	INT NOT NULL
Client ID	INT NOT NULL
Meeting Date	DATE NOT NULL
Agenda	VARCHAR (30) NOT NULL

Client Review Table

Review ID	PK
Client ID	FK, PK
Rating	DECIMAL(3,2) NOT NULL
Comment	VARCHAR(10)
Feedback	VARCHAR (100)

Designer Review Table

Review ID	PK
Designer ID	FK, PK
Rating	DECIMAL(3,2) NOT NULL
Comment	VARCHAR(10)
Feedback	VARCHAR (100)

Supplier Review Table

Review ID	PK
Supplier ID	FK, PK
Rating	DECIMAL(3,2) NOT NULL
Comment	VARCHAR(10)

Feedback	VARCHAR (100)
----------	---------------

Order Table

Order ID	PK
Client ID	FK
Designer ID	FK
Address ID	FK
OrderStatus	VARCHAR (10)
Delivery Date	DATE

Order Products Table

Order ID	FK, PK
Product ID	FK, PK
Quantity	INT

Product Table

Product ID	PK
Supplier ID	FK
Description	VARCHAR (500) NOT NULL
Cost	INT NOT NULL
Quantity	INT NOT NULL

Discounts Table

Discount Code	VARCHAR(10) PK
Discount Amount	DECIMAL(10,2) NOT NULL

Financial Table

Tracking ID	PK
OrderID	FK
Project ID	FK
Payment Amount	INT NOT NULL
Payment Date	DATE NOT NULL
Payment Method	VARCHAR (5)
Discount Code	VARCHAR(5)
Total Amount	INT NOT NULL
Status	VARCHAR (10)

CREATE TABLE SCRIPTS FOR ALL TABLES ARE AS BELOW :

- 1. CREATE TABLE R_Address (**
AddressID BIGINT PRIMARY KEY,
Street VARCHAR(10),
City VARCHAR(10),
State VARCHAR(10),
Zipcode INT
);
- 2. CREATE TABLE R_Client (**
ClientID bigint PRIMARY KEY,
Name VARCHAR(20),
Contact BIGINT,
Budget INT,
AddressID INT REFERENCES R_Address(AddressID)
);
- 3. CREATE TABLE R_Designer (**
DesignerID bigint PRIMARY KEY,
Name VARCHAR(20),
Contact BIGINT,
Portfolio_Link VARCHAR(15),
Experience INT,
Rating DECIMAL(3, 2)
);
- 4. CREATE TABLE R_Project (**
ProjectID bigint PRIMARY KEY,
DesignerID INT REFERENCES R_Designer(DesignerID),
ClientID INT REFERENCES R_Client(ClientID),
Start_Date DATE,
End_Date DATE,
Status varchar(20) CHECK (Status in('Pending', 'In Progress', 'Completed')),
Cost DECIMAL(10, 2)
);
- 5. CREATE TABLE R_Appointment (**
AppointmentID bigint PRIMARY KEY,
DesignerID INT REFERENCES R_Designer(DesignerID),
ClientID INT REFERENCES R_Client(ClientID),
Meeting_Date DATE,
Agenda VARCHAR
);

```
6. CREATE TABLE R_Client_Review (
    ReviewID bigint,
    ClientID INT REFERENCES R_Client(ClientID),
    Rating DECIMAL(3, 2),
    Comment VARCHAR(20),
    Feedback VARCHAR(200),
    PRIMARY KEY (ReviewID, ClientID)
);

7. CREATE TABLE R_Designer_Review (
    ReviewID bigint,
    DesignerID INT REFERENCES R_Designer(DesignerID),
    Rating DECIMAL(3, 2),
    Comment VARCHAR (20),
    Feedback VARCHAR (200),
    PRIMARY KEY (ReviewID, DesignerID)
);

8. CREATE TABLE R_Supplier_Review (
    ReviewID SERIAL,
    SupplierID INT,
    Rating DECIMAL(3, 2),
    Comment VARCHAR (20),
    Feedback VARCHAR (200),
    PRIMARY KEY (ReviewID, SupplierID)
);

9. CREATE TABLE R_Order (
    OrderID bigint PRIMARY KEY,
    ClientID INT REFERENCES R_Client(ClientID),
    DesignerID INT REFERENCES R_Designer(DesignerID),
    AddressID INT REFERENCES R_Address(AddressID),
    Delivery_Date DATE,
    OrderCost INT,
    OrderStatus VARCHAR(50)
);

10. CREATE TABLE R_Order_Products (
    OrderID INT REFERENCES R_Order(OrderID),
    ProductID INT,
    Quantity INT,
    PRIMARY KEY (OrderID, ProductID)
);
```

11. CREATE TABLE R_Product (
ProductID BIGINT PRIMARY KEY,
SupplierID INT,
Description VARCHAR,
Cost DECIMAL(10, 2),
Quantity INT
);

12. CREATE TABLE R_Suppliers (
SupplierID BIGINT PRIMARY KEY,
Name VARCHAR(20),
Contact BIGINT,
Portfolio_Link VARCHAR(15),
Experience INT,
Rating DECIMAL(3, 2)
);

13. CREATE TABLE Discounts (
Discount_Code varchar(255) PRIMARY KEY,
Discount_Amount decimal(10,2) NOT NULL
);

14. CREATE TABLE Financial_Transactions (
Tracking_ID serial PRIMARY KEY,
OrderID int NULL,
ProjectID int NULL,
Payment_Amount decimal(10,2) NOT NULL,
Payment_Date date NOT NULL,
Payment_Method varchar(50) NOT NULL,
Discount_Code varchar(255) NULL REFERENCES Discounts(Discount_Code),
Total_Amount decimal(10,2) NOT NULL,
Status varchar(50) NOT NULL
);

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree structure of database objects like Schemas, Tables, and Views. The right pane is the Query Editor, showing a SQL script for creating tables. The script includes comments and code for Address, Client, and Designer tables, along with a CREATE TABLE statement for a table named 'CREATE TABLE'. The status bar at the bottom indicates the query was successful.

```
-- Address table
-- CREATE TABLE R_Address (
--     AddressID bigint PRIMARY KEY,
--     Street VARCHAR(255) NOT NULL,
--     City VARCHAR(100) NOT NULL,
--     State VARCHAR(100) NOT NULL,
--     Zipcode VARCHAR(20) NOT NULL
-- );
-- 
-- -- Client table
-- CREATE TABLE R_Client (
--     ClientID bigint PRIMARY KEY,
--     Name VARCHAR(255) NOT NULL,
--     Contact VARCHAR(255) NOT NULL,
--     Budget DECIMAL(10, 2) NOT NULL,
--     Feedback TEXT,
--     AddressID INT REFERENCES R_Address(AddressID) ON DELETE SET NULL
-- );
-- 
-- -- Designer table
-- CREATE TABLE R_Designer
```

INSERT SCRIPTS FOR TABLES ARE AS BELOW:

Client Table:

```
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (10, 'Bat',  
2465267816, 7601.12, 56);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (2372,  
'Melessa', 2571788833, 6564.13, 5775);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (755,  
'Pip', 7722803317, 9193.68, 7212);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (298,  
'Kristoffer', 5126648894, 8585.65, 5775);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (17,  
'Nevsa', 1154230900, 9575.81, 9125);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (81,  
'Terri', 2845678983, 9567.39, 68);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1160,  
'Corinna', 2207856423, 3215.42, 544);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (423,  
'Tatum', 9973337880, 7596.00, 9125);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (680,  
'Con', 3069115006, 7840.97, 60);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (13,  
'Elwin', 3491683459, 9558.41, 0);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (69,  
'Jamison', 9199449880, 1759.95, 8);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (443,  
'Nananne', 2181415511, 8995.96, 9740);  
-- Continue with the rest of your INSERT queries  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (793,  
'Bernie', 3412394148, 5622.57, 817);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (4522,  
'Ealasaid', 8724536584, 6384.19, 8);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (744,  
'Orly', 7724299216, 9040.24, 7778);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (57,  
'Frannie', 6576668431, 9914.52, 97248);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (2112,  
'Spenser', 7959705073, 9799.87, 2568);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (22,  
'Shandee', 6394811979, 7888.80, 4);  
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1985,  
'Nils', 9396041295, 1635.45, 4786);
```

```
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (496, 'Maisey', 1098555607, 1175.89, 2);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (2430, 'Elinor', 4655982950, 1988.85, 0);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (6352, 'Aguistin', 5226108154, 7024.57, 2320);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1234, 'Alison', 9269771235, 4380.87, 5);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (452, 'Amil', 6756395093, 1447.15, 60);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1486, 'Leif', 6299256530, 5505.67, 5);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (2345, 'Aube', 5642587312, 6253.78, 68);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (3456, 'Joelie', 8962706031, 1869.32, 693);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (4043, 'Davie', 7326980893, 8036.23, 704);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (601, 'Rosabella', 5091974264, 9567.86, 54);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (60, 'Aurelia', 4403679927, 9799.38, 45);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1009, 'Gearard', 1237035354, 9371.09, 5096);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (137, 'Roi', 5131807000, 9716.14, 69816);
-- Continue with the rest of your INSERT queries
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (398, 'Cull', 2029381248, 2161.23, 50);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (574, 'Bellina', 5058809114, 3231.85, 9540);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (59, 'Theo', 3753770891, 4315.73, 69);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (6752, 'Konstantin', 8496919054, 9341.48, 18474);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (3050, 'Candida', 4852682801, 3459.14, 7);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (400, 'Timofei', 7048673390, 3637.61, 45);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (682, 'Christen', 3882196340, 8970.53, 9125);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (4567, 'Rab', 3691680257, 7491.45, 92971);
```

```
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (492, 'Annelise', 7217409203, 5406.19, 54);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (571, 'Armin', 2199683181, 1129.36, 4786);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (961, 'Chelsey', 9649465128, 6284.65, 60);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (85, 'Adelbert', 6949337135, 1467.42, 0);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (128, 'Marisa', 8137980472, 4060.37, 704);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (224, 'Clevie', 4198052274, 9915.24, 7622);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (532, 'Magda', 6645450255, 5302.30, 04);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (16, 'Barnebas', 4247542872, 6684.09, 9740);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (9, 'Derril', 6545642917, 5146.38, 4);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (6100, 'Octavius', 9947337492, 8804.35, 496);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1133, 'Veradis', 6433409657, 2557.57, 50);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (799, 'Aliza', 7974987131, 6338.79, 5775);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (984, 'Arnuad', 3475015175, 8794.16, 49442);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (3299, 'Tamara', 4626407985, 9028.90, 0);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (1980, 'Olly', 3461098970, 6104.40, 5);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (24, 'Sheryl', 9676859263, 4813.31, 7);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (3627, 'Celina', 2192990455, 3488.19, 4592);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (117, 'Brennen', 4266627028, 5468.54, 04);
INSERT INTO R_Client (ClientID, Name, Contact, Budget, AddressID) VALUES (3301, 'Madison', 2599002720, 3371.54, 0203);
```

Products Table :

```
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(342, 45625, 'Rhododendron macrophyllum D. Don ex G. Don', 872, 5);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(341, 75007, 'Greyia Hook. & Harv.', 12878, 24);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(259, 60416, 'Orthotrichum flowersii Vitt', 12352, 14);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(276, 24099, 'Camissonia minor (A. Nelson) P.H. Raven', 15563, 9);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(307, 3310, 'Rhus kearneyi F.A. Barkley', 10490, 27);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(236, 22140, 'Quercus michauxii Nutt.', 13561, 7);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(321, 98934, 'Cyrtandra tintinnabula Rock', 5872, 28);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(287, 22306, 'Teesdalia coronopifolia (Bergeret) Thell.', 21128, 3);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(215, 21335, 'Dicerandra frutescens Shinners ssp. modesta R.B. Huck', 19099, 15);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(295, 142906, 'Raillardella (A. Gray) Benth.', 10296, 15);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(180, 88983, 'Luzula congesta (Thuill.) Lej.', 8252, 7);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(177, 28270, 'Potentilla atrosanguinea Lodd. ex D. Don', 2021, 1);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(175, 96426, 'Helichrysum Mill.', 11784, 21);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(2410, 21335, 'Arenaria fendleri A. Gray', 17189, 4);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(193, 73927, 'Aspicilia rosulata KÃ¶rb.', 15778, 20);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(212, 156496, 'Bryum rutilans Brid.', 17527, 18);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(270, 29226, 'Halenia Borkh.', 16286, 4);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(303, 22140, 'Helenium flexuosum Raf.', 2586, 8);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(1756, 83660, 'Heuchera americana L. var. americana', 21876, 23);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(2192, 23044, 'Angelica archangelica L.', 4384, 8);  
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES  
(2761, 73927, 'Paspalum urvillei Steud.', 17276, 24);
```

INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(326, 47975, 'Usnea glabrescens (Nyl. ex Vain.) Vain.', 14336, 12);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(289, 153446, 'Chorizanthe xanti S. Watson', 21344, 20);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(343, 30512, 'Althaea cannabina L.', 20527, 15);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(1917, 123005, 'Duboisia R. Br.', 12749, 11);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(255, 97390, 'Guillemina densa (Humb. & Bonpl. ex Schult.) Moq. var. aggregata Uline & Bray', 11982, 27);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(294, 21335, 'Eryngium aromaticum Baldw.', 10969, 20);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(224, 75935, 'Oryza sativa L.', 20523, 12);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(2114, 97390, 'Boerhavia wrightii A. Gray', 5275, 30);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(200, 49404, 'Joinvillea Gaudich. ex Brongn. & Gris', 3533, 21);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(196, 95332, 'Melicope kavaiensis (H. Mann) T.G. Hartley & B.C. Stone', 17183, 10);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(281, 1855, 'Carex manhartii Bryson', 7730, 22);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(320, 92163, 'Juncus alpinoarticulatus Chaix', 9972, 7);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(324, 55298, 'Eriosorus FÃ©', 3242, 4);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(309, 55298, 'Lesquerella arctica (Wormsk. ex Hornem.) S. Watson', 5914, 27);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(304, 75985, 'Peperomia sintenisii C. DC.', 17140, 25);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(232, 75007, 'Lupinus excubitus M.E. Jones var. austromontanus (A. Heller) C.P. Sm.', 7985, 27);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(248, 132798, 'Juniperus communis L. var. communis L. [excluded]', 11528, 11);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(347, 95402, 'Fulglesia fulgens (Sw.) Elenkin', 18947, 18);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(338, 73927, 'Artemisia scopulorum A. Gray', 11861, 14);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(315, 95332, 'Thelypodium sagittatum (Nutt. ex Torr. & A. Gray) Endl. ex Walp. ssp. ovalifolium (Rydb.) Al-Shehbaz', 1047, 22);

INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(214, 3310, 'Verbena polystachya Kunth', 10687, 22);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(292, 78465, 'Pottia arizonica Wareh. var. mucronulata Wareh.', 18986, 13);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(257, 86434, 'Mimulus latidens (A. Gray) Greene', 16957, 3);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(2817, 19139, 'Tetraclea coulteri A. Gray', 22309, 6);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(229, 88839, 'Myosotis discolor Pers.', 19463, 25);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(240, 95332, 'Silene virginica L.', 6172, 13);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(194, 51135, 'Lesquerella —maxima (Rollins) Rollins', 6955, 9);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(344, 60416, 'Astragalus pycnostachyus A. Gray var. pycnostachyus', 9655, 3);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(197, 126699, 'Carex williamsii Britton', 12184, 11);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(245, 19139, 'Smilax jamesii G. Wallace', 19365, 11);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(195, 1855, 'Arthonia albovirescens Nyl.', 5070, 21);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(302, 153446, 'Nothochelone (A. Gray) Straw', 13720, 20);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(164, 153446, 'Anthoxanthum aristatum Boiss.', 14814, 5);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(166, 45625, 'Lemna L.', 17047, 17);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(256, 75985, 'Cyperus sphacelatus Rottb.', 16370, 27);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(310, 30512, 'Paronychia virginica Spreng.', 13825, 11);
INSERT INTO R_Product (ProductID, SupplierID, Description, Cost, Quantity) VALUES
(2516, 23044, 'Pseudoleskea incurvata (Hedw.) Loeske var. tenuiretis (Culm.) H.A. Crum,
Steere & L.E. Anderson', 18522, 23);

Address Table :

```

pgAdmin 4
File Object Tools Help
Object Explorer Dashboard Properties SQL Statistics Dependencies Dependents Processes Address.sql
Interior_DBMS /postgres@PostgreSQL 14
Query History
Query
30 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('10', 'Tomscot', 'Myrtle Beach', 'South Carolina', '410-08-5099');
31 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('48627', 'Loftsgordon', 'Panama City', 'Florida', '472-25-4237');
32 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('86459', 'Rieder', 'Washington', 'District of Columbia', '308-54-7591');
33 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('5', 'Grasskamp', 'Lynn', 'Massachusetts', '632-61-7086');
34 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('485', 'Veith', 'Corona', 'California', '497-18-3505');
35 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('613', 'Hanson', 'Omaha', 'Nebraska', '156-13-3504');
36 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('208', 'West', 'San Francisco', 'California', '516-36-8631');
37 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('59943', 'Melvin', 'Bowie', 'Maryland', '386-62-8561');
38 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('4341', 'Hagan', 'Rockford', 'Illinois', '132-57-5268');
39 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('331', 'Northland', 'North Little Rock', 'Arkansas', '760-34-2310');
40 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('644', 'Eagan', 'Kansas City', 'Kansas', '799-26-9116');
41 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('77', 'Shoshone', 'San Bernardino', 'California', '386-46-0851');
42 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('27', 'Dottie', 'Sterling', 'Virginia', '285-93-9166');
43 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('5548', 'Farmco', 'Shawnee Mission', 'Kansas', '407-09-1308');
44 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('94', 'Scofield', 'Memphis', 'Tennessee', '459-09-3915');
45 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('68', 'Hintze', 'Muskegon', 'Michigan', '688-25-6155');
46 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('7647', 'Lake View', 'Bronx', 'New York', '744-12-2676');
47 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('30', 'Crescent Oaks', 'Hot Springs National Park', 'Arkansas', '887-43-2777');
48 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('277', 'Hagan', 'Oakland', 'California', '494-65-2148');
49 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('9', 'Forster', 'Lexington', 'Kentucky', '695-65-7322');
50 INSERT INTO R_Address (AddressID, Street, City, State, Zipcode) values ('9', 'Forster', 'Lexington', 'Kentucky', '695-65-7322');

Data Output Messages Notifications
INSERT 0 1
Query returned successfully in 54 msec.

Total rows: 0 of 0 Query complete 00:00:00.054
Ln 49, Col 77
File saved successfully.
ENG IN 11:02 07-11-2023

```

Designer Table :

```

pgAdmin 4
File Object Tools Help
Object Explorer Dashboard Properties SQL Statistics Dependencies Dependents Processes Address.sql* Interior_DBMS /postgres@PostgreSQL 14*
Query History
Query
1 INSERT INTO R_Designer (DesignerID, Name, Contact, Portfolio_Link, Experience, Rating)
2 VALUES
3 (1101, 'Designer1', 1234567890, 'https://portfolio1.com', 5, 4.75),
4 (1102, 'Designer2', 9876543210, 'https://portfolio2.com', 8, 4.90),
5 (1103, 'Designer3', 5551234567, 'https://portfolio3.com', 3, 4.50),
6 (1104, 'Designer4', 1237894560, 'https://portfolio4.com', 6, 4.60),
7 (1105, 'Designer5', 7894561230, 'https://portfolio5.com', 4, 4.70),
8 (1106, 'Designer6', 6547891230, 'https://portfolio6.com', 7, 4.80),
9 (1107, 'Designer7', 1239876540, 'https://portfolio7.com', 2, 4.40),
10 (1108, 'Designer8', 5555555555, 'https://portfolio8.com', 9, 4.95),
11 (1109, 'Designer9', 6666666666, 'https://portfolio9.com', 3, 4.55),
12 (1110, 'Designer10', 7777777777, 'https://portfolio10.com', 5, 4.70),
13 (1111, 'Designer11', 8888888888, 'https://portfolio11.com', 7, 4.85),
14 (1112, 'Designer12', 9999999999, 'https://portfolio12.com', 6, 4.75),
15 (1113, 'Designer13', 1618101810, 'https://portfolio13.com', 4, 4.60),
16 (1114, 'Designer14', 1212121212, 'https://portfolio14.com', 8, 4.80),
17 (1115, 'Designer15', 1313131313, 'https://portfolio15.com', 5, 4.70),
18 (1116, 'Designer16', 1414141414, 'https://portfolio16.com', 6, 4.75),
19 (1117, 'Designer17', 1515151515, 'https://portfolio17.com', 3, 4.55),
20 (1118, 'Designer18', 1616161616, 'https://portfolio18.com', 9, 4.90);

Data Output Messages Notifications
INSERT 0 20
Query returned successfully in 124 msec.

Total rows: 0 of 0 Query complete 00:00:00.124
Ln 1, Col 15
File saved successfully.
ENG IN 11:21 07-11-2023

```

Project Table :

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database schema with the **Tables (13)** node expanded, revealing tables like r_address, r_appointment, r_client, etc.
- SQL Editor:** Contains the following SQL code:

```

1  INSERT INTO R_Project (ProjectID, DesignerID, ClientID, Start_date, End_date, Status, Cost)
2  VALUES
3  (2101, 1101, 101, '2023-01-15', '2023-02-28', 'Pending', 5000.00),
4  (2102, 1102, 102, '2023-02-18', '2023-03-20', 'In Progress', 7200.00),
5  (2103, 1103, 103, '2023-03-05', '2023-04-15', 'Pending', 5500.00),
6  (2104, 1104, 104, '2023-04-01', '2023-05-10', 'Completed', 6800.00),
7  (2105, 1105, 105, '2023-05-20', '2023-06-30', 'In Progress', 4500.00),
8  (2106, 1106, 106, '2023-06-15', '2023-07-25', 'Pending', 6900.00),
9  (2107, 1107, 107, '2023-07-10', '2023-08-15', 'Completed', 8500.00),
10 (2108, 1108, 108, '2023-08-05', '2023-09-12', 'In Progress', 7100.00),
11 (2109, 1109, 109, '2023-09-01', '2023-10-22', 'Pending', 5900.00),
12 (2110, 1110, 110, '2023-10-15', '2023-11-18', 'Completed', 6300.00),
13 (2111, 1111, 111, '2023-11-10', '2023-12-25', 'Pending', 7400.00),
14 (2112, 1112, 112, '2023-12-05', '2024-01-30', 'In Progress', 7200.00),
15 (2113, 1113, 113, '2024-01-15', '2024-02-25', 'Completed', 5800.00),
16 (2114, 1114, 114, '2024-02-10', '2024-03-20', 'In Progress', 6600.00),
17 (2115, 1115, 115, '2024-03-05', '2024-04-15', 'Pending', 7200.00),
18 (2116, 1116, 116, '2024-04-01', '2024-05-10', 'Completed', 6700.00),
19 (2117, 1117, 117, '2024-05-20', '2024-06-30', 'Pending', 5500.00),
20 (2118, 1118, 118, '2024-06-15', '2024-07-25', 'In Progress', 7200.00),

```
- Data Output:** Shows the result of the query: **INSERT 0 20**.
- Messages:** Shows a success message: **Query returned successfully in 84 msec.**
- Notifications:** None.
- System Bar:** Shows the status bar with **Ln 5, Col 67**, **11:27**, **07-11-2023**, and system icons.

Appointment Table :

The screenshot shows the pgAdmin 4 interface with the following details:

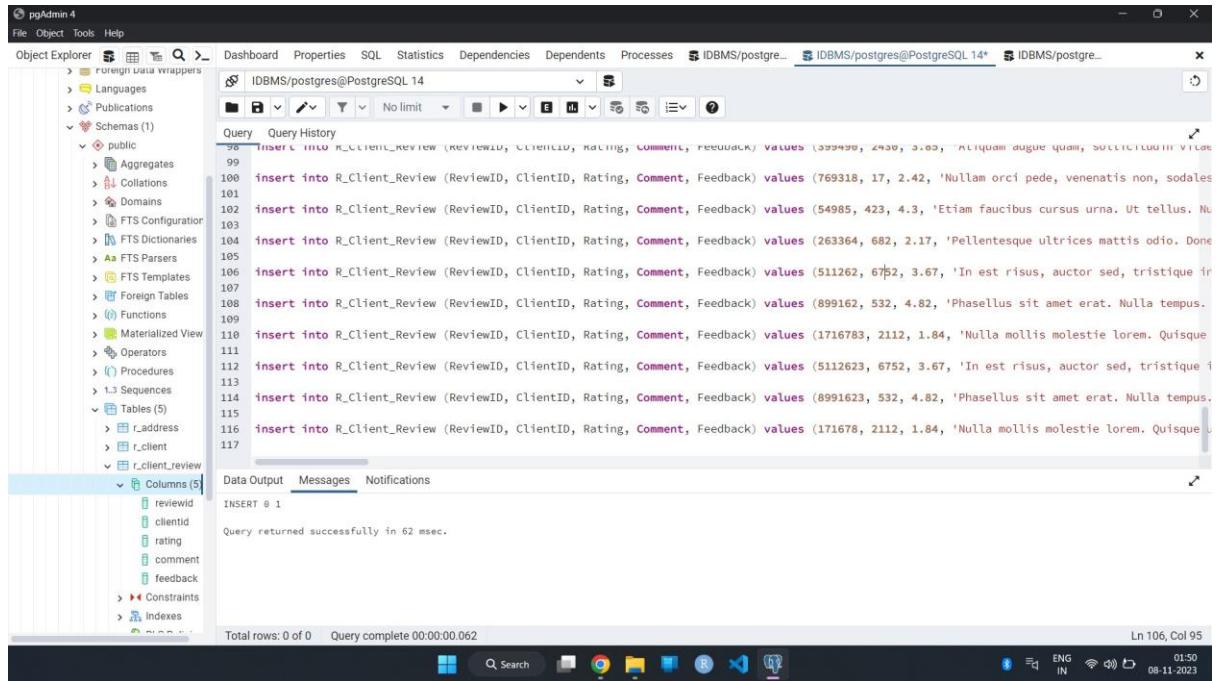
- Object Explorer:** Shows the database schema with the **Tables (5)** node expanded, revealing tables like r_address, r_client, etc.
- SQL Editor:** Contains the following SQL code:

```

1  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (1, 10, 398, '2023-11-03', 'Martes pennanti');
2  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (2, null, 793, '2023-11-04', 'Dromaeus novaehollandiae');
3  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (3, 49, 60, '2023-11-10', 'Marmota caligata');
4  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (4, 26, 4043, '2023-11-01', 'Lasiocnora parahybana');
5  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (5, 37, 59, '2023-11-03', 'Odocoileus hemionus');
6  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (6, null, 2430, '2023-11-09', 'Felis wiedii or Leopoldina wiedii');
7  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (7, null, 682, '2023-11-08', 'Acotophilonis africanus');
8  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (8, 20, null, '2023-11-05', 'Toxostoma curvirostre');
9  INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (9, null, 423, '2023-11-03', 'Rangifer tarandus');
10 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (10, null, 4522, '2023-11-03', 'Oryctelus fulgidus');
11 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (11, 15, 137, '2023-11-01', 'Funambulus pennatus');
12 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (12, null, 85, '2023-11-08', 'Milvago chimachima');
13 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (13, 50, null, '2023-11-09', 'Dicrurus adsimilis');
14 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (14, 45, 1133, '2023-11-08', 'Bubalush arnee');
15 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (15, null, 9, '2023-11-03', 'Myiarchus tuberculifer');
16 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (16, null, 1160, '2023-11-08', 'Spermophilus parryi');
17 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (17, null, null, '2023-11-10', 'Chlidonias leucopterus');
18 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (18, 38, 680, '2023-11-02', 'Ovis mustimon');
19 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (19, null, 532, '2023-11-10', 'Ciconia episcopus');
20 INSERT INTO R_Appointment (AppointmentID, DesignerID, ClientID, Meeting_Date, Agenda) VALUES (20, null, 744, '2023-11-08', 'Meleagris gallopavo')

```
- Data Output:** Shows the result of the query: **INSERT 0 20**.
- Messages:** Shows a success message: **Query returned successfully in 54 msec.**
- Notifications:** None.
- System Bar:** Shows the status bar with **Ln 44, Col 107**, **01:24**, **08-11-2023**, and system icons.

Client Review Table :



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with Schemas (1), Tables (5), and r_client_review table.
- Query Editor:** Displays the SQL code for creating the R_Client_Review table and inserting sample data. The table has columns: reviewid, clientid, rating, comment, feedback.
- Data Output:** Shows the successful insertion of 1 row.
- Messages:** Displays the message "Query returned successfully in 62 msec."
- System Bar:** Shows the status bar with "Ln 106, Col 95".

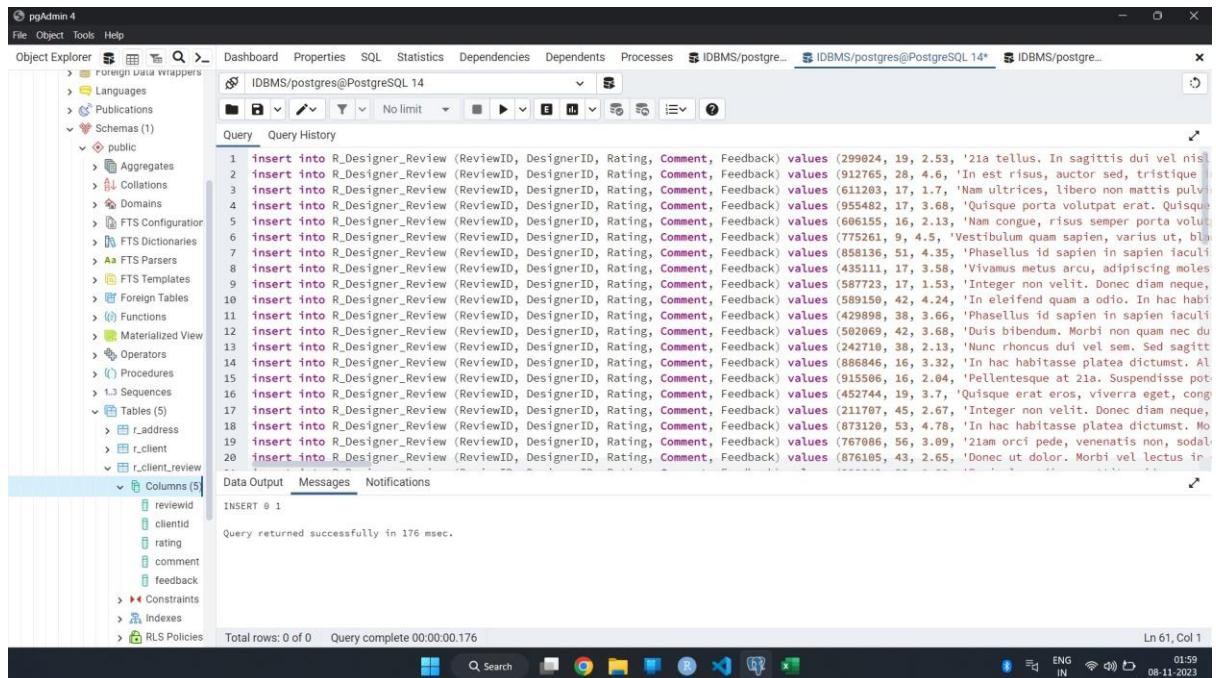
```

CREATE TABLE R_Client_Review (
    reviewid INT,
    clientid INT,
    rating INT,
    comment TEXT,
    feedback TEXT
);

INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (399490, 2430, 2.63, 'Aliquam augue quam, sollicitudin vel');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (769318, 17, 2.42, 'Nullam orci pede, venenatis non, sodales');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (54985, 423, 4.3, 'Etiam faucibus cursus urna. Ut tellus. Nu');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (263364, 682, 2.17, 'Pellentesque ultrices mattis odio. Done');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (511262, 6752, 3.67, 'In est risus, auctor sed, tristique in');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (899162, 532, 4.82, 'Phasellus sit amet erat. Nulla tempus.');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (1716783, 2112, 1.84, 'Nulla mollis molestie lorem. Quisque');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (5112623, 6752, 3.67, 'In est risus, auctor sed, tristique in');
INSERT INTO R_Client_Review (ReviewID, ClientID, Rating, Comment, Feedback) VALUES (8991623, 532, 4.82, 'Phasellus sit amet erat. Nulla tempus.');

```

Designer Review :



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with Schemas (1), Tables (5), and r_client_review table.
- Query Editor:** Displays the SQL code for creating the R_Designer_Review table and inserting sample data. The table has columns: reviewid, clientid, rating, comment, feedback.
- Data Output:** Shows the successful insertion of 1 row.
- Messages:** Displays the message "Query returned successfully in 176 msec."
- System Bar:** Shows the status bar with "Ln 61, Col 1".

```

CREATE TABLE R_Designer_Review (
    reviewid INT,
    clientid INT,
    rating INT,
    comment TEXT,
    feedback TEXT
);

INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (299024, 19, 2.53, 'Vestibulum quam sapien, varius ut, blandit non, euismod id, ligula. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (912765, 28, 4.6, 'In est risus, auctor sed, tristique in');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (611203, 17, 1.7, 'Nam ultrices, libero non mattis pulvinar, nulla non metus auctor nec');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (955482, 17, 3.68, 'Quisque porta voluptatibus erat. Quisque');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (606155, 16, 2.13, 'Nam congue, risus semper porta volutpat, quam sapien porttitor');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (775261, 9, 4.5, 'Vestibulum quam sapien, varius ut, blandit non, euismod id, ligula. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (858136, 51, 4.35, 'Phasellus id sapien in sapien iaculis. Nam ultrices, libero non mattis pulvinar, nulla non metus auctor nec');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (435111, 17, 3.58, 'Vivamus metus arcu, adipiscing molestie');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (587723, 17, 1.53, 'Integer non velit. Donec diam neque, blandit non, euismod id, ligula. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (589150, 42, 4.24, 'In eleifend quam a odio. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (429898, 38, 3.66, 'Phasellus id sapien in sapien iaculis. Nam');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (502069, 42, 3.68, 'Duis bibendum. Morbi non quam nec dui');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (242710, 38, 2.13, 'Nunc rhoncus dui vel sem. Sed sagittis');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (886846, 16, 3.32, 'In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (915506, 16, 2.84, 'Pellentesque at 21a. Suspidisse potest');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (452744, 19, 3.7, 'Quisque erat eros, viverra eget, congue');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (211707, 45, 2.67, 'Integer non velit. Donec diam neque, blandit non, euismod id, ligula. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (873120, 53, 4.78, 'In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (767086, 56, 3.09, 'Vestibulum quam sapien, varius ut, blandit non, euismod id, ligula. In hac habitasse platea dictumst. Ali');
INSERT INTO R_Designer_Review (ReviewID, DesignerID, Rating, Comment, Feedback) VALUES (876105, 43, 2.65, 'Donec ut dolor. Morbi vel lectus in');

```

Order Products :

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the Object Explorer with two servers: PostgreSQL 10 and PostgreSQL 14. Under PostgreSQL 14, the 'public' schema is selected, showing various objects like Aggregates, Collations, Domains, FTS Configuration, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized View, Operators, Procedures, Sequences, and Tables (14). The right pane contains a SQL editor tab titled 'IDBMS/postgres@PostgreSQL 14*'. The query window shows a series of INSERT statements into the 'R_Order_Products' table, with a total of 60 rows inserted. The status bar at the bottom indicates 'Total rows: 60 of 60' and 'Query complete 00:00:00.133'. A green message bar at the bottom right says '✓ Query returned successfully in 133 msec.'

```
42 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (58, 289, 18);
43 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (39, 343, 21);
44 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (28, 315, 30);
45 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (7, 315, 11);
46 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (42, 186, 19);
47 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (22, 240, 19);
48 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (53, 321, 18);
49 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (1, 338, 5);
50 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (15, 303, 18);
51 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (9, 197, 18);
52 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (20, 281, 20);
53 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (43, 208, 15);
54 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (59, 307, 17);
55 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (37, 342, 10);
56 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (8, 256, 25);
57 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (42, 307, 5);
58 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (52, 344, 22);
59 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (34, 321, 26);
60 INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (55, 195, 20);
61 |
```

Discounts Table :

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the Object Explorer with two servers: PostgreSQL 10 and PostgreSQL 14. Under PostgreSQL 14, the 'public' schema is selected, showing various objects like Aggregates, Collations, Domains, FTS Configuration, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized View, Operators, Procedures, Sequences, and Tables (14). The right pane contains a SQL editor tab titled 'IDBMS/postgres@PostgreSQL 14*'. The query window shows a series of INSERT statements into the 'Discounts' table, with a total of 60 rows inserted. The status bar at the bottom indicates 'Total rows: 60 of 60' and 'Query complete 00:00:00.075'. A green message bar at the bottom right says '✓ Query returned successfully in 75 msec.'

```
1 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#6ac', 163.22);
2 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#b01', 174.7);
3 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#e46', 225.06);
4 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#268', 122.04);
5 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#7bc', 935.09);
6 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#a49', 596.79);
7 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#38e', 670.65);
8 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#f1e', 56.72);
9 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#9dc', 63.23);
10 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#a01', 421.27);
11 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#d48', 821.66);
12 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#3d2', 570.36);
13 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#831', 166.26);
14 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#b22', 580.06);
15 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#ed6', 831.73);
16 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#d1d', 284.64);
17 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#cc8', 455.91);
18 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#17b', 192.75);
19 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#0ba', 588.93);
20 INSERT INTO Discounts (Discount_Code, Discount_Amount) VALUES ('#e62', 154.62);
```

Financial Transaction Table :

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with nodes like `r_address`, `r_appointment`, `r_client`, `r_client_review`, `r_review`, `r_reviewid`, `r_clientid`, `r_rating`, `r_comment`, `r_feedback`, `r_constraints`, `r_indexes`, `r_RLS_Policies`, `r_rules`, `r_triggers`, `r_designer`, `r_designer_rev1`, `r_order`, `r_order_produc`, `r_product`, `r_project`, `r_supplier_rev1`, `r_suppliers`, `r_trigger_functions`, `r_types`, `r_views`, `r_subscriptions`, `postgres`, `Login/Group Roles`, and `Tablespaces`.
- Query Editor:** Displays several SQL `INSERT INTO` statements for the `Financial_Transactions` table. The first few statements are:

```
VALUES (25, 45, 21, 29012, '2023-05-16', 'Credit Card', '#7bc', 110466, 'Successful');
INSERT INTO Financial_Transactions (Tracking_ID, OrderID, ProjectID, Payment_Amount, Payment_Date, Payment_Method, Discount_Code, Total_Amount, VALUES (26, 60, 3, 10093, '2023-01-22', 'UPI', '#ed6', 96668, 'Unsuccessful');
INSERT INTO Financial_Transactions (Tracking_ID, OrderID, ProjectID, Payment_Amount, Payment_Date, Payment_Method, Discount_Code, Total_Amount, VALUES (27, 22, 14, 70355, '2022-12-16', 'Netbanking', '#836', 84018, 'Unsuccessful');
```

- Data Output:** Shows the result of the query with 61 rows inserted.
- System Bar:** Includes icons for search, refresh, and various system status indicators (e.g., battery, signal, date).

Order Products Table :

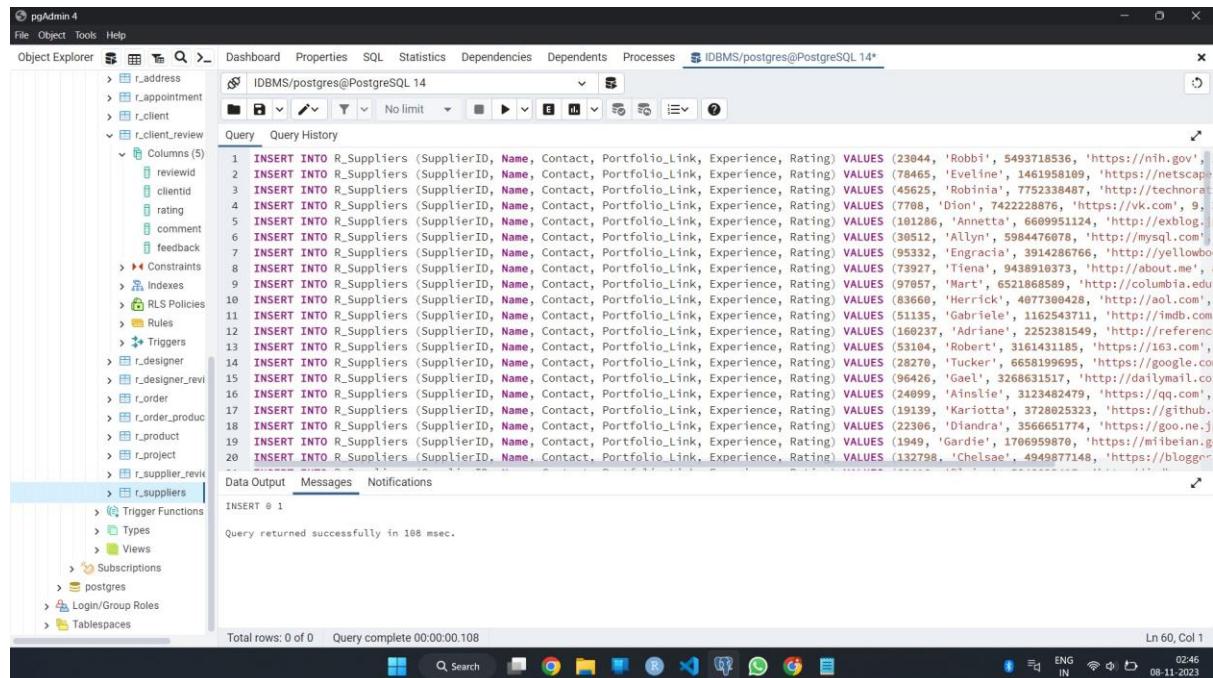
The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with nodes like `Servers (2)`, `PostgreSQL 10`, `PostgreSQL 14`, `Databases (2)`, `IDBMS`, `Casts`, `Catalogs`, `Event Triggers`, `Extensions`, `Foreign Data Wrappers`, `Languages`, `Publications`, `Schemas (1)`, `public`, `Aggregates`, `Collations`, `Domains`, `FTS Configuration`, `FTS Dictionaries`, `FTS Parsers`, `FTS Templates`, `Foreign Tables`, `Functions`, `Materialized View`, `Operators`, `Procedures`, `Sequences`, `Tables (14)`, `discounts`, and `financial_trans`.
- Query Editor:** Displays several SQL `INSERT INTO` statements for the `R_Order_Products` table. The first few statements are:

```
VALUES (58, 289, 18);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (30, 343, 21);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (28, 315, 30);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (7, 315, 11);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (42, 186, 19);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (22, 246, 19);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (53, 321, 18);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (1, 338, 5);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (15, 303, 18);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (9, 197, 18);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (20, 281, 20);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (43, 206, 15);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (59, 307, 17);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (37, 342, 10);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (8, 256, 25);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (42, 307, 5);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (52, 344, 22);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (34, 321, 26);
INSERT INTO R_Order_Products (OrderID, ProductID, Quantity) VALUES (55, 195, 20);
```

- Data Output:** Shows the result of the query with 60 rows inserted.
- System Bar:** Includes icons for search, refresh, and various system status indicators (e.g., battery, signal, date).

Supplier Table :



```

pgAdmin 4
File Object Tools Help
Object Explorer Dashboard Properties SQL Statistics Dependencies Dependents Processes IDBMS/postgres@PostgreSQL 14*
Query Query History
1 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (23044, 'Robbi', 5493718536, 'https://nih.gov', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
2 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (78465, 'Eveline', 1461958109, 'https://netscape.com', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
3 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (45625, 'Robinia', 7752338487, 'http://technorati.com', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
4 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (7708, 'Dion', 7422228876, 'https://vk.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
5 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (101286, 'Annetta', 6609951124, 'http://exblog.jp', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
6 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (30512, 'Allyn', 5984476078, 'http://mysq.com', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
7 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (95332, 'Engracia', 3914286766, 'http://yellowbot.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
8 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (73927, 'Tiena', 9438910373, 'http://about.me', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
9 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (97057, 'Mart', 6521868589, 'http://columbia.edu', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
10 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (83666, 'Herrick', 40773080428, 'http://aol.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
11 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (51135, 'Gabriele', 1162543711, 'http://imdb.com', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
12 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (106237, 'Adriane', 2252381549, 'http://reference.com', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
13 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (53104, 'Robert', 3161431185, 'https://163.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
14 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (28270, 'Tucker', 6658199695, 'https://google.com', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
15 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (96426, 'Gael', 3288631517, 'http://dailymail.co.uk', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
16 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (24099, 'Ainslie', 3123482479, 'https://qq.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
17 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (19139, 'Kariotta', 3728025323, 'https://github.com', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
18 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (22306, 'Diandra', 3566651774, 'https://goo.ne.jp', 1.1, 'Whiskey Oscar Romeo Bravo November India Yankee Papa Oscar Golf Delta Lima')
19 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (1949, 'Gardie', 1706959870, 'https://microsoft.com', 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
20 INSERT INTO R_Suppliers (SupplierID, Name, Contact, Portfolio_Link, Experience, Rating) VALUES (132798, 'Chelsae', 4949877148, 'https://blogspot.com', 1.1, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')

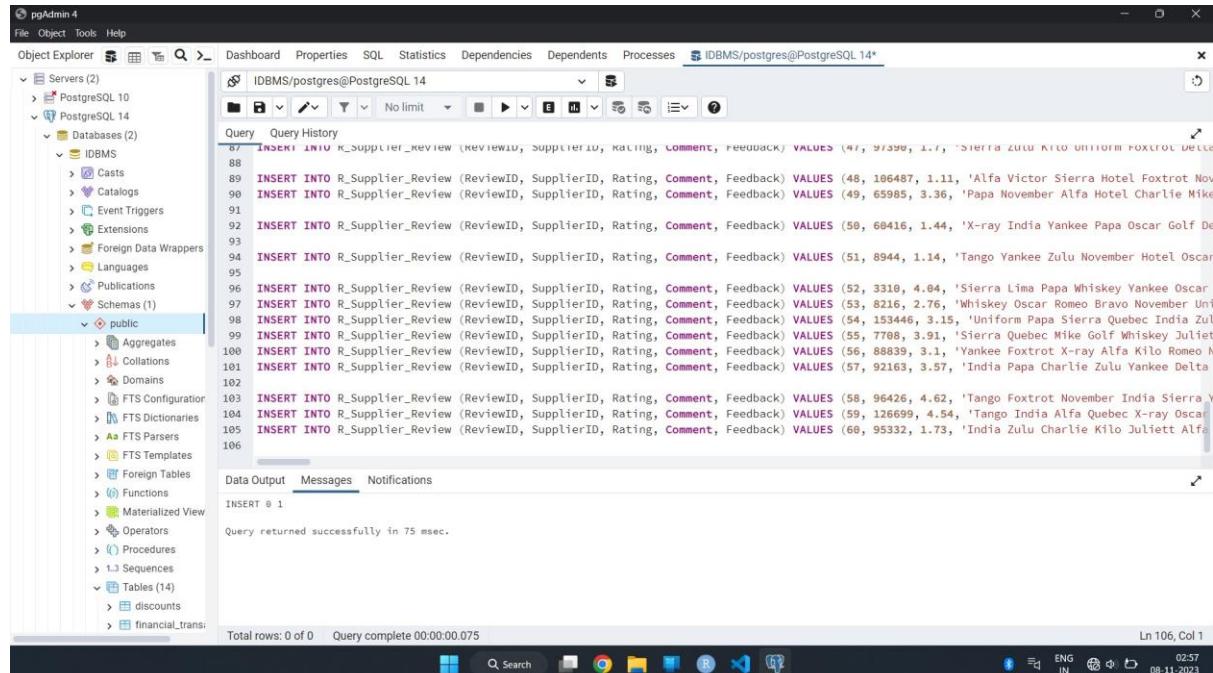
Data Output Messages Notifications
INSERT 0 1

Query returned successfully in 188 msec.

Total rows: 0 of 0 Query complete 00:00:00.108 Ln 60, Col 1

```

Supplier Review Table :



```

pgAdmin 4
File Object Tools Help
Object Explorer Dashboard Properties SQL Statistics Dependencies Dependents Processes IDBMS/postgres@PostgreSQL 14*
Query Query History
87 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (47, 97390, 1.1, 'Sierra Zulu Kilo Uniform Foxrot Oscar Charlie Mike')
88 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (48, 106487, 1.11, 'Alfa Victor Sierra Hotel Foxtrot November India Yankee Papa Oscar Golf Delta Lima')
89 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (49, 65985, 3.36, 'Papa November Alfa Hotel Charlie Mike')
90 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (50, 60416, 1.44, 'IX-ray India Yankee Papa Oscar Golf Delta Lima')
91 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (51, 8944, 1.14, 'Tango Yankee Zulu November Hotel Oscar')
92 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (52, 3310, 4.04, 'Sierra Lima Papa Whiskey Yankee Oscar')
93 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (53, 8216, 2.76, 'Whiskey Oscar Romeo Bravo November Uni')
94 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (54, 153446, 3.15, 'Uniform Papa Sierra Quebec India Zulu')
95 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (55, 7708, 3.91, 'Sierra Quebec Mike Golf Whiskey Juliet')
96 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (56, 88839, 3.1, 'Yankee Foxtrot X-ray Alfa Kilo Romeo Hotel Oscar')
97 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (57, 92163, 3.57, 'India Papa Charlie Zulu Yankee Delta')
98 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (58, 96426, 4.62, 'Tango Foxtrot November India Sierra')
99 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (59, 126699, 4.54, 'Tango India Alfa Quebec X-ray Oscar')
100 INSERT INTO R_Supplier_Review (ReviewID, SupplierID, Rating, Comment, Feedback) VALUES (60, 95332, 1.73, 'India Zulu Charlie Kilo Juliett Alfa')

Data Output Messages Notifications
INSERT 0 1

Query returned successfully in 75 msec.

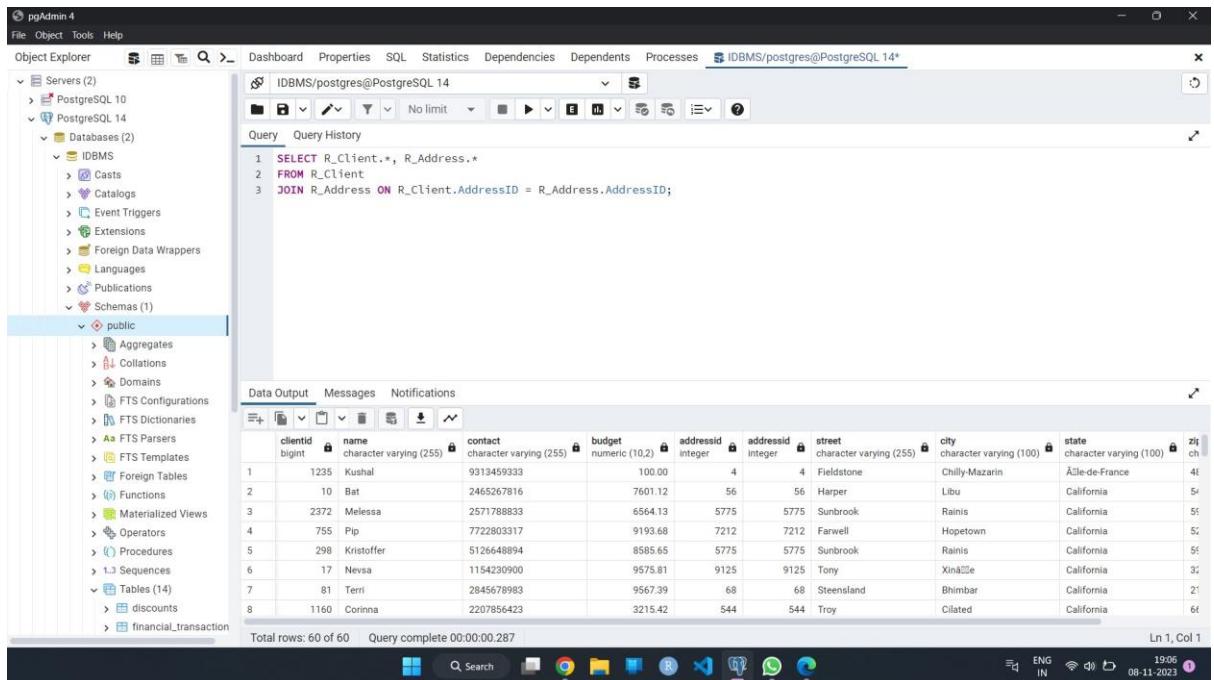
Total rows: 0 of 0 Query complete 00:00:00.075 Ln 106, Col 1

```

QUERIES IN PLAIN ENGLISH AND IN SQL:

1. Retrieve all client information along with their address:

Sql: SELECT R_Client.*, R_Address.*
 FROM R_Client
 JOIN R_Address ON R_Client.AddressID = R_Address.AddressID;



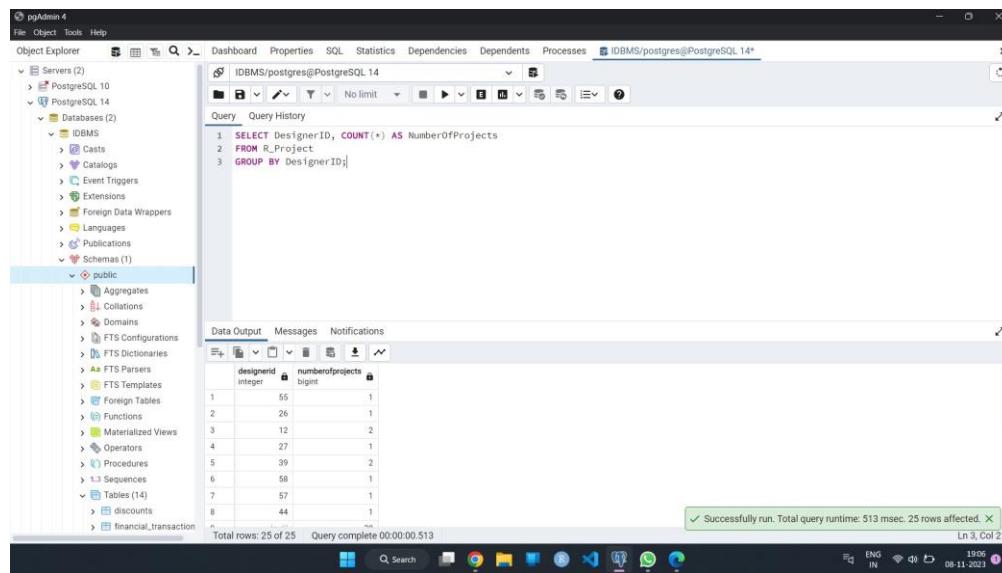
The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with servers, databases, and tables.
- Query Editor:** Contains the SQL query:


```
1 SELECT R_Client.*, R_Address.*  
2 FROM R_Client  
3 JOIN R_Address ON R_Client.AddressID = R_Address.AddressID;
```
- Data Output:** Displays the results of the query as a table. The columns are clientid, name, contact, budget, addressid, street, city, state, and zip. The data includes rows for clients like Kushal, Bat, and Kristoffer, each associated with a specific address.
- System Bar:** Shows the system tray with icons for battery, signal, and date/time (08-11-2023).

2. Find the total number of projects for each designer:

Sql: SELECT DesignerID, COUNT(*) AS NumberOfProjects FROM R_Project GROUP BY DesignerID;



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with servers, databases, and tables.
- Query Editor:** Contains the SQL query:


```
1 SELECT DesignerID, COUNT(*) AS NumberOfProjects  
2 FROM R_Project  
3 GROUP BY DesignerID;
```
- Data Output:** Displays the results of the query as a table. The columns are designerid and numberofprojects. The data shows the count of projects for each designer ID from 1 to 8.
- System Bar:** Shows the system tray with icons for battery, signal, and date/time (08-11-2023). A message at the bottom right indicates the query was successfully run with a runtime of 513 msec and 25 rows affected.

3. List all projects that are currently active:

Sql : SELECT * FROM R_Project WHERE Status = 'Active';

The screenshot shows the pgAdmin 4 interface. In the Object Explorer, the 'Tables (14)' node is selected. In the main query editor, the following SQL query is run:

```
1 SELECT * FROM R_Project WHERE Status = 'In Progress';
```

The results are displayed in a Data Output tab, showing 10 rows of project data. The columns are: projectid [PK] bigint, designerid integer, clientid integer, start_date date, end_date date, status character varying(20), and cost numeric(10,2). The 'status' column consistently shows 'In Progress'.

projectid [PK]	designerid	clientid	start_date	end_date	status	cost
11	[null]	2430	2022-09-10	2023-10-22	In Progress	[null]
12	3	1133	2021-04-16	2023-01-28	In Progress	361.51
13	[null]	17	2022-03-13	2023-05-04	In Progress	517.05
14	[null]	[null]	2021-03-04	2023-04-21	In Progress	[null]
15	42	799	2020-09-08	2023-08-19	In Progress	970.47
16	51	60	2020-11-26	2023-07-28	In Progress	[null]
17	5	81	2019-04-24	2023-08-12	In Progress	687.05
18	4	60	2022-02-07	2023-07-06	In Progress	[null]
19	5	55	2022-03-11	2023-08-08	In Progress	68.00

4. Get the average rating of all designers:

Sql : SELECT AVG(Rating) AS AverageRating FROM R_Designer;

The screenshot shows the pgAdmin 4 interface. In the Object Explorer, the 'Tables (14)' node is selected. In the main query editor, the following SQL query is run:

```
1 SELECT ROUND(AVG(Rating), 2) AS AverageRating
2 FROM R_Designer;
3 |
```

The results are displayed in a Data Output tab, showing a single row with the average rating. The column is labeled 'averaging' and has a value of 2.68.

averaging
2.68

At the bottom right, there are three green success messages indicating the query was run successfully:

- Successfully run. Total query runtime: 151 msec. 1 rows affected.
- Query returned successfully in 53 msec.
- Query returned successfully in 60 msec.

5. Show the top 5 designers by experience:

Sql: SELECT * FROM R_Designer ORDER BY Experience DESC LIMIT 5;

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables.
- Query Editor:** Contains the following SQL code:

```
1 SELECT *
2 FROM R_Designer
3 ORDER BY Experience DESC
4 LIMIT 5;
```
- Data Output:** Displays the results of the query, showing 5 rows of data:| | designerid | bigin | name | contact | portfolio_link |
| --- | --- | --- | --- | --- | --- |
| 1 | 40 | Margeaux | 8724596954 | http://typepad.com/augue/aliquam.json?mus=nec&vivamus=sem&vestibulum=duis&sagittis=aliquam&sapien=convallis&cum=nunc&sociis=pr |
| 2 | 37 | Reggie | 3497830752 | https://webeden.co.uk/sodales/scoleisque/mauris/sit/amet.png?velit=tincidunt&eu=iacus&est=at&congue=velit&elementum=vivamus&in=vel |
| 3 | 22 | Bendite | 6339463309 | https://state.gov/mattis/egestas/metus/aeonian/fer |
| 4 | 5 | Chilton | 4427041958 | http://un.org/nei/vulputate/nonummy/maecenas/tincidunt/iacus.jsp?turpis=aliquet&nec=ac&eu=turpis&leo=neque&mauris=&non=lacinia& |
| 5 | 41 | Nickola | 1518160968 | http://arstechnica.com/accumsan/tortor/quis/turpis.jpg?montes=cras&nascetur=pellentesque/ridiculus=volutpat&mus=dui&vivamus=mae |
- System Bar:** Shows the status bar with "Total rows: 5 of 5" and "Query complete 00:00:00.167".
- Bottom Right:** Shows system icons for battery, signal, and date/time (08-11-2023, 19:18).

6. List all orders with their corresponding client names:

Sql :SELECT R_Order.* , R_Client.Name FROM R_Order JOIN R_Client ON R_Order.ClientID = R_Client.ClientID;

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables.
- Query Editor:** Contains the following SQL code:

```
1 SELECT R_Order.* , R_Client.Name FROM R_Order
2 JOIN R_Client ON R_Order.ClientID = R_Client.ClientID;
```
- Data Output:** Displays the results of the query, showing 55 rows of data:| | orderid | clientid | designerid | addressid | delivery_date | ordercost | orderstatus | name |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1486 | 31 | 60 | 2022-12-30 | 418624 | Cancelled | Leif |
| 2 | 2 | 1009 | 16 | 9740 | 2023-07-19 | 543536 | Processing | Gearard |
| 3 | 3 | 60 | 55 | 704 | 2023-05-08 | 2464557 | Shipped | Aurelia |
| 4 | 4 | 60 | 33 | 92971 | 2023-08-09 | 1768256 | Cancelled | Aurelia |
| 5 | 5 | 6100 | [null] | 49442 | 2023-09-03 | 1627578 | Shipped | Octavius |
| 6 | 6 | 2372 | [null] | 4 | 2023-07-04 | 1638037 | Shipped | Melessa |
| 7 | 7 | 4043 | [null] | 4 | 2023-02-16 | 1682476 | Cancelled | Davie |
| 8 | 9 | 224 | [null] | 0 | 2023-03-26 | 2214032 | Shipped | Clevie |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
- System Bar:** Shows the status bar with "Total rows: 55 of 55" and "Query complete 00:00:00.138".
- Bottom Right:** Shows system icons for battery, signal, and date/time (08-11-2023, 19:18).

7. Find the total cost of projects for each client:

Sql : SELECT ClientID, SUM(Cost) AS TotalProjectCost
FROM R_Project
GROUP BY ClientID;

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree view of database objects including Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables. The 'r_project' table is selected. The right pane contains a query editor window with the following content:

```
1 SELECT ClientID, SUM(Cost) AS TotalProjectCost FROM R_Project GROUP BY ClientID;
```

The Data Output tab shows the results of the query:

	clientid	totalprojectcost
1	69	670.50
2	6752	538.30
3	60	[null]
4	3627	148.24
5	17	517.05
6	22	547.24
7	680	963.07
8	3301	692.99

Total rows: 53 of 53 Query complete 00:00:00.108

At the bottom right of the pgAdmin window, there is a system tray showing the date (08-11-2023), time (19:19), and battery status.

8. Display all products that have been ordered more than once:

Sql: SELECT ProductID, SUM(Quantity) AS TotalOrdered
FROM R_Order_Products
GROUP BY ProductID
HAVING SUM(Quantity) > 1;

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'r_product' table is selected.
- Query Editor:** Contains the following SQL code:

```
1 SELECT ProductID, SUM(Quantity) AS TotalOrdered
2 FROM R_Order_Products
3 GROUP BY ProductID
4 HAVING SUM(Quantity) > 1;
5
```
- Data Output:** Displays the results of the query in a table format. The columns are 'productid' and 'totalordered'. The data is as follows:

productid	totalordered
1	344
2	347
3	294
4	229
5	321
6	245
7	240
8	303

Message bar: Successfully run. Total query runtime: 88 msec. 39 rows affected.

System tray: ENG IN 19:20 08-11-2023

9. Retrieve the list of all appointments for the next 7 days:

Sql: SELECT *
FROM R_Appointment
WHERE Meeting_Date BETWEEN CURRENT_DATE AND CURRENT_DATE +
INTERVAL '7 days';

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'r_appointment' table is selected.
- Query Editor:** Contains the following SQL code:

```
1 SELECT *
2 FROM R_Appointment
3 WHERE Meeting_Date BETWEEN CURRENT_DATE AND CURRENT_DATE + INTERVAL '7 days';
4
```
- Data Output:** Displays the results of the query in a table format. The columns are 'appointmentid', 'designerid', 'clientid', 'meeting_date', and 'agenda'. The data is as follows:

appointmentid	designerid	clientid	meeting_date	agenda
1	3	49	2023-11-10	Marmota caligata
2	6	[null]	2430	Felis wiedi or Leopardus wiedi
3	7	[null]	682	Actophilomys africanus
4	12	[null]	85	Milvago chimachima
5	13	50	[null]	Dicurus adsimilis
6	14	45	1134	Bubalus arnee
7	16	[null]	1160	Spermophilus parryi
8	17	[null]	[null]	Chlidonias leucopterus

Message bar: Successfully run. Total query runtime: 88 msec. 22 rows affected.

System tray: ENG IN 19:20 08-11-2023

10. Show the average rating of each supplier:

Sql: SELECT SupplierID, AVG(Rating) AS AverageRating
FROM R_Supplier_Review
GROUP BY SupplierID;

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like Collations, Domains, FTS Configurations, etc. The right pane contains a query editor and a data output window.

Query Editor:

```
1 SELECT SupplierID, ROUND(AVG(Rating),2) AS AverageRating
2 FROM R_Supplier_Review
3 GROUP BY SupplierID;
4
```

Data Output:

SupplierID	AverageRating
45678	4.20
95402	1.14
73786	4.08
78956	4.70
75007	1.68
12345	4.30
55298	3.30
3310	3.56

Total rows: 41 of 41 Query complete 00:00:00.159

11. Get the list of clients with their total spent on orders:

Sql: SELECT R_Client.ClientID, R_Client.Name, SUM(R_Order.Cost) AS TotalSpent
FROM R_Client
JOIN R_Order ON R_Client.ClientID = R_Order.ClientID
GROUP BY R_Client.ClientID;

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like Collations, Domains, FTS Configurations, etc. The right pane contains a query editor and a data output window.

Query Editor:

```
1 SELECT R_Client.ClientID, R_Client.Name, SUM(R_Order.ordercost) AS TotalSpent
2 FROM R_Client
3 JOIN R_Order ON R_Client.ClientID = R_Order.ClientID
4 GROUP BY R_Client.ClientID;
```

Data Output:

ClientID	Name	TotalSpent
984	Arnuad	23152
574	Bellina	931048
1009	Gearard	543536
3299	Tamara	308587
10	Bat	1651604
443	Nannane	241949
6352	Aguistin	1841954
799	Aliza	1478754

Total rows: 51 of 51 Query complete 00:00:00.079

12. List the projects that went over budget:

Sql: SELECT *

FROM R_Project

JOIN R_Client ON R_Project.ClientID = R_Client.ClientID

WHERE R_Project.Cost > R_Client.Budget;

The screenshot shows the pgAdmin 4 interface. The left sidebar is titled 'Object Explorer' and lists various database objects under 'Tables (14)'. The main area contains a SQL editor with the following query:

```
1 SELECT *
2 FROM R_Project
3 JOIN R_Client ON R_Project.ClientID = R_Client.ClientID
4 WHERE R_Project.Cost > R_Client.Budget;
5
```

Below the query, the 'Data Output' tab displays the results in a table:

	projectid	designerid	clientid	start_date	end_date	status	cost	clientid	name	contact	budget
1	23	38	682	2018-06-18	2023-09-18	Pending	9000.00	682	Christen	3882196340	8970.53

A green success message at the bottom right of the data output area says 'Data saved successfully.'.

13. Display all the products with less than 10 items in stock:

Sql: SELECT *

FROM R_Product

WHERE Quantity < 10;

The screenshot shows the pgAdmin 4 interface. The left sidebar is titled 'Object Explorer' and lists various database objects under 'Tables (14)'. The main area contains a SQL editor with the following query:

```
1 SELECT *
2 FROM R_Product
3 WHERE Quantity < 10;
4
```

Below the query, the 'Data Output' tab displays the results in a table:

	productid	supplierid	description	cost	quantity
1	342	45625	Rhododendron macrophyllum D. Don ex G. Don	872.00	5
2	276	24099	Camissonia minor (A. Nelson) P.H. Raven	15563.00	9
3	236	22140	Quercus michauxii Nutt.	13561.00	7
4	287	22306	Teesdalia coronopifolia (Bergeret) Thell.	21128.00	3
5	180	88983	Luzula congesta (Thunb.) Lej.	8252.00	7
6	177	28270	Potentilla atrosanguinea Lodd. ex D. Don	2021.00	1
7	2410	21335	Arenaria fenderi A. Gray	17189.00	4
8	270	29226	Habenaria Borkh.	16286.00	4

A green success message at the bottom right of the data output area says 'Successfully run. Total query runtime: 330 msec. 17 rows affected.'

14. Show the details of the latest 10 transactions:

```
Sql : SELECT *
FROM Financial_Transactions
ORDER BY Payment_Date DESC
LIMIT 10;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- File**, **Object**, **Tools**, **Help** menu at the top.
- Object Explorer** sidebar on the left listing various database objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Dashboard**, **Properties**, **SQL**, **Statistics**, **Dependencies**, **Dependents**, **Processes** tabs at the top of the main area.
- IDBMS/postgres@PostgreSQL 14*** connection selected in the top bar.
- Query History** tab selected in the main area.
- Query** tab content:

```
1 SELECT *
2 FROM Financial_Transactions
3 ORDER BY Payment_Date DESC
4 LIMIT 10;
```
- Data Output** tab selected in the bottom navigation bar.
- Table** showing the results of the query:

	tracking_id	orderid	projectid	payment_amount	payment_date	payment_method	discount_code	total_amount	status
1	15	36	30	109096.00	2023-11-03	COD	#c50	100585.00	Unsuccessful
2	2	34	9	84875.00	2023-10-30	Credit Card	#6ac	53458.00	Successful
3	51	50	27	27491.00	2023-10-29	Netbanking	#05b	91802.00	Successful
4	41	19	47	66385.00	2023-10-15	UPI	#ce3	57295.00	Unsuccessful
5	50	38	15	106746.00	2023-10-12	Netbanking	#722	65546.00	Unsuccessful
6	58	21	39	102976.00	2023-10-07	Debit Card	#ee3	73065.00	Successful
7	55	29	11	99668.00	2023-10-03	COD	#9c1	66572.00	Unsuccessful
8	60	29	19	108639.00	2023-09-29	Debit Card	#6a9	76254.00	Successful
...

- Total rows: 10 of 10
- Query complete 00:00:00.118
- System status bar at the bottom: ENG IN 21:39 08-11-2023

15. List all designers who have a rating above 4.0:

Sql :

```
SELECT *
FROM R_Designer
WHERE Rating > 4.0;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database schema with various objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Contains the SQL query:

```
1 SELECT *
2 FROM R_Designer
3 WHERE Rating > 4.0;
```
- Data Output:** Displays the results of the query:

	designerid	name	contact	portfolio_link
1	1	Babette	9977402197	http://opensource.org/tincidunt/lacus.jsp?rtor=cum&duis=sociis&mattis=natoque&egestas=penatibus&metus=et&aenean=magnis
2	5	Chilton	4427041958	http://un.org/nisi/vulputate/nonummy/maccenas/tincidunt/lacus.jsp?utris&neque&mauris&non=lacini
3	12	Karola	3011965408	https://paypal.com/aliquam/augue/quam/sollicitudin/vitae/consectetur/egert.png?vel=libero&accumsan=nam&tellus=dui&nisi=proin&eu=le
4	15	Kellia	3230841267	https://booton.com/nec/nisi/vulputate/nonummy/mauris/plates.json?veit=dolor&ne=congue/non&nisi=lacinia=volutpat=egert&unc=tinc
5	17	Krysta	3183541470	https://nymag.com/suspendisse/lacinia/nulla.xml?le=ut&vel=congue&sc=sed&congue=nisl&fusce=smobi=iacus&dpibus=orci&nulla=lu
6	20	Alma	3146951975	https://nih.gov/massa/vulputat/convallis/morbi/odio/curabitur.jsp?aliquam=fusce&non=magna&mauris&ac=proin&in=a&porta=ligula&ut=ul
7	23	Cammie	1058625123	https://state.gov/vel/augue/vestibulum.json?n=aliquam&felis=duis&sodales=ac&honcus=nibh&sed=fermentum&vestibulum=vel&tincidunt
8	36	Sari	9663924198	https://princeton.edu/in/felis/donec/sempre/sapi
- Messages:** Shows the message "Successfully run. Total query runtime: 191 msec. 11 rows affected."
- System Bar:** Shows the taskbar with various icons and the system clock indicating 08-11-2023 21:41.

16. Retrieve the total revenue from all completed projects:

Sql :

```
SELECT SUM(Cost) AS TotalRevenue
FROM R_Project
WHERE Status = 'Completed';
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database schema with various objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14).
- Query Editor:** Contains the SQL query:

```
1 SELECT SUM(Cost) AS TotalRevenue
2 FROM R_Project
3 WHERE Status = 'Completed';
```
- Data Output:** Displays the results of the query:

	totalrevenue
1	5851.61
- Messages:** Shows the message "Successfully run. Total query runtime: 127 msec. 1 rows affected."
- System Bar:** Shows the taskbar with various icons and the system clock indicating 08-11-2023 21:42.

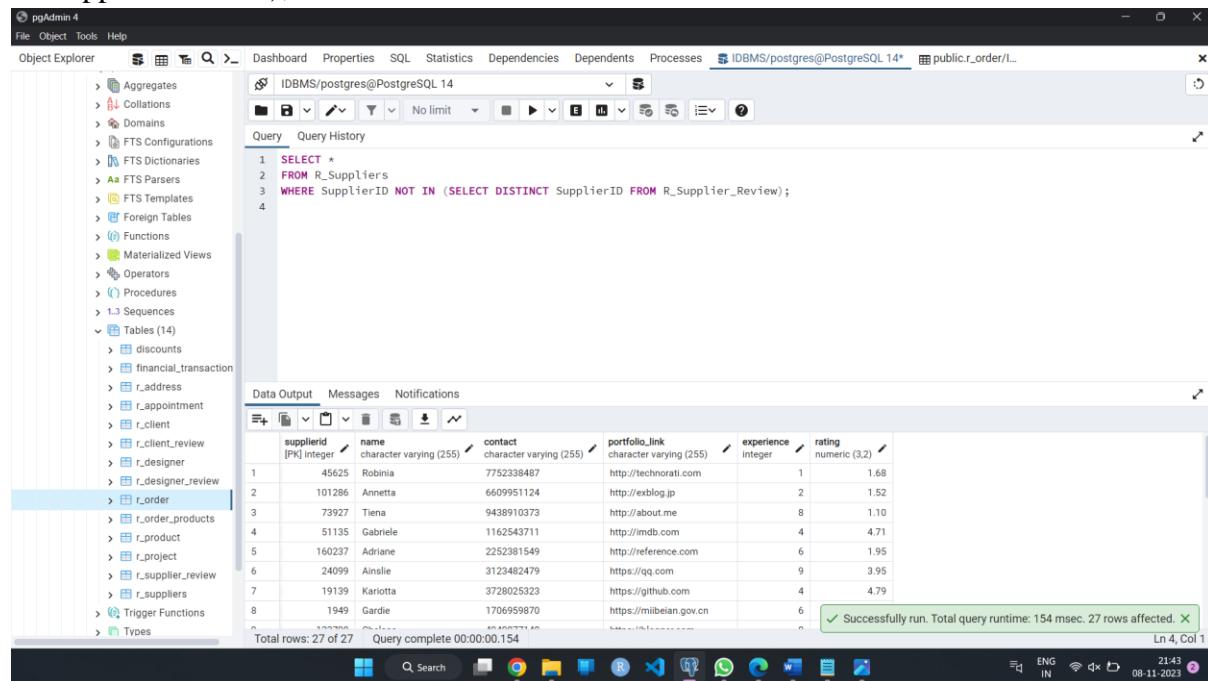
17. List all suppliers that have not been reviewed yet:

Sql:

```
SELECT *
```

```
FROM R_Suppliers
```

```
WHERE SupplierID NOT IN (SELECT DISTINCT SupplierID FROM  
R_Supplier_Review);
```



The screenshot shows the pgAdmin 4 interface with a query editor and a data output window.

Query Editor:

```
1 SELECT *\n2 FROM R_Suppliers\n3 WHERE SupplierID NOT IN (SELECT DISTINCT SupplierID FROM R_Supplier_Review);\n4
```

Data Output:

	SupplierID [PK] Integer	Name character varying (255)	Contact character varying (255)	Portfolio_Link character varying (255)	Experience integer	Rating numeric (3,2)
1	45625	Robinia	7752338487	http://technorati.com	1	1.68
2	101286	Annetta	6609951124	http://exblog.jp	2	1.52
3	73927	Tierna	9438910373	http://about.me	8	1.10
4	51135	Gabriele	1162543711	http://imdb.com	4	4.71
5	160237	Adriane	2252381549	http://reference.com	6	1.95
6	24099	Ainslie	3123482479	https://qq.com	9	3.95
7	19139	Karriota	3728025323	https://github.com	4	4.79
8	1949	Gardie	1706959870	https://milibeian.gov.cn	6	n

Total rows: 27 of 27 Query complete 00:00:00.154

LN 4, Col 1

Successfully run. Total query runtime: 154 msec. 27 rows affected.

18. Show the count of projects each client has initiated:

Sql:

```
SELECT ClientID, COUNT(*) AS ProjectCount  
FROM R_Project  
GROUP BY ClientID;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is the Object Explorer, displaying various database objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and 14 Tables. The 'r_order' table is selected. The main pane contains a SQL editor with the following query:

```
1 SELECT ClientID, COUNT(*) AS ProjectCount  
2 FROM R_Project  
3 GROUP BY ClientID;
```

The Data Output tab shows the results:

clientid	projectcount
1	69
2	6752
3	60
4	3627
5	17
6	22
7	680
8	3301

Below the table, the status bar indicates "Successfully run. Total query runtime: 161 msec. 53 rows affected." and "Ln 4, Col 1".

19. Find the total revenue generated per designer from projects:

Sql:

```
SELECT R_Designer.Name, SUM(R_Project.Cost) AS TotalRevenue  
FROM R_Project  
JOIN R_Designer ON R_Project.DesignerID = R_Designer.DesignerID  
GROUP BY R_Designer.Name;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is the Object Explorer, displaying various database objects like FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized View, Operators, Procedures, Sequences, and 14 Tables. The 'r_suppliers' table is selected. The main pane contains a SQL editor with the following query:

```
1 SELECT R_Designer.Name, SUM(R_Project.Cost) AS TotalRevenue  
2 FROM R_Project  
3 JOIN R_Designer ON R_Project.DesignerID = R_Designer.DesignerID  
4 GROUP BY R_Designer.Name;
```

The Data Output tab shows the results:

name	totalrevenue
Curtis	270.95
Bard	554.34
Fedora	1181.09
Reeva	266.90
Karola	354.90
Tessy	223.59
Reggie	[null]
Enrica	970.47

Below the table, the status bar indicates "Total rows: 24 of 24" and "Query complete 00:00:00.131".

20. Determine the average project duration for each designer:

Sql:

```
SELECT R_Designer.Name, AVG(R_Project.End_Date - R_Project.Start_Date) AS AverageDuration
FROM R_Project
JOIN R_Designer ON R_Project.DesignerID = R_Designer.DesignerID
GROUP BY R_Designer.Name;
```

The screenshot shows the pgAdmin 4 interface. The left sidebar is the Object Explorer, displaying various database objects like tables, functions, and procedures. The main area is a query editor window with the following SQL code:

```
1 SELECT R_Designer.Name, AVG(R_Project.End_Date - R_Project.Start_Date) AS AverageDuration
2 FROM R_Project JOIN R_Designer ON R_Project.DesignerID = R_Designer.DesignerID
3 GROUP BY R_Designer.Name;
```

Below the code, the results are displayed in a table:

	name	averageduration
1	Curtis	143.00000000000000
2	Band	969.00000000000000
3	Fedora	566.00000000000000
4	Reeva	947.00000000000000
5	Karola	877.00000000000000
6	Tessy	608.00000000000000
7	Reggie	1779.00000000000000
8	Enrica	1075.00000000000000

Total rows: 24 of 24 Query complete 00:00:00.096 Ln 2, Col 16

21. List clients with the number of appointments and projects they have:

Sql:

```
SELECT R_Client.Name,
       (SELECT COUNT(*) FROM R_Appointment WHERE R_Appointment.ClientID = R_Client.ClientID) AS AppointmentCount,
       (SELECT COUNT(*) FROM R_Project WHERE R_Project.ClientID = R_Client.ClientID)
AS ProjectCount
FROM R_Client;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized View, Operators, Procedures, Sequences, Tables (14), Triggers, Functions, Types, Views, Subscriptions, and Tablespace.
- Query Editor:** Contains the SQL query provided above.
- Data Output:** Displays the results of the query in a tabular format.
- Results:**

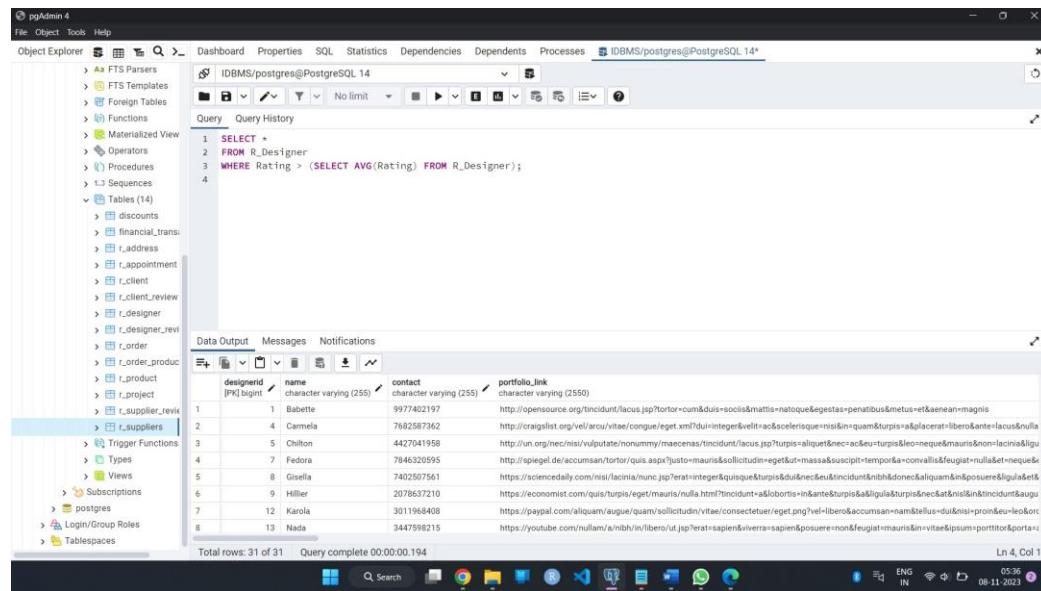
	name	character varying (255)	appointmentcount	bigint	projectcount	bigint
1	Bat			1		1
2	Melessa			1		2
3	Pip			1		1
4	Kristoffer			1		1
5	Neva			2		2
6	Terri			1		1
7	Corinna			1		1
8	Tatum			1		1
...	...			*		*

Total rows: 59 of 59 Query complete 00:00:00.126

22. Retrieve designers with a rating above the average rating of all designers:

Sql:

```
SELECT *
FROM R_Designer
WHERE Rating > (SELECT AVG(Rating) FROM R_Designer);
```



The screenshot shows the pgAdmin 4 interface with a query editor containing the following SQL code:

```
1 SELECT *
2   FROM R_Designer
3 WHERE Rating > (SELECT AVG(Rating) FROM R_Designer);
4
```

The results pane displays the following data:

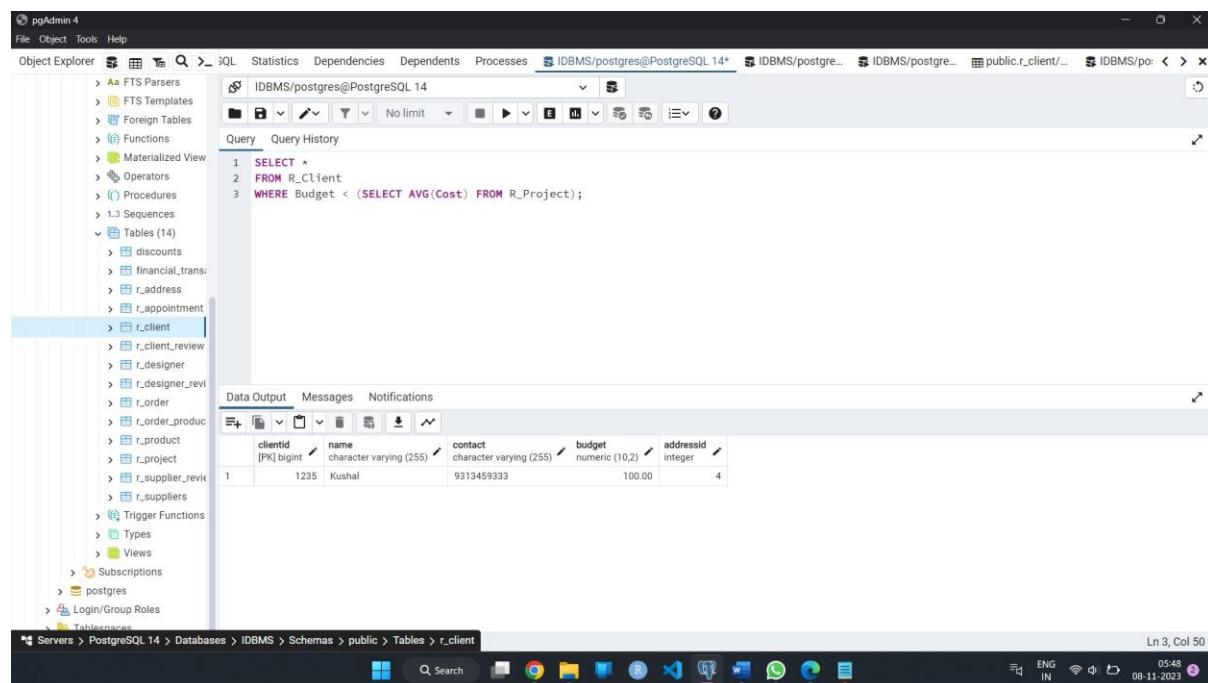
designerid	name	contact	portfolio_link
1	Babette	9977420197	http://opensource.org/trac/dt/lacus.jsp?rtor=cum&duis=soci&matis=natoque&egestas=penatibus&metus=et&aenean=magnis
2	Carmela	7682587362	http://craiglist.org/vell/arcu/inter/congue/egert.xml?dui=integer&velit=acc&celerisque=most&im&quam&turpis=ad&laceat=libero&ante=iacus&nulla
3	Chilton	4427041958	http://un.org/nec/mis/vulputate/nonummy/maecenas/incipit/lacus.jsp?tupis=aliqu&necc&ac&eius&turpis&leo=neque&mauris=knob=laicis&ligu
4	Fedora	7846320595	http://spiegel.de/acumsan/tortor/ut/quis.aspx?justo=maurus&sollicitudin=egert&et=massa&asciipit=temper&a;convallis&feugiat=nulla&et=
5	Gisella	7402597051	https://sciencedaily.com/rest/facina/munc.jsp?erat=integer&quique=turpis&dui=nece&eius&incipit/ultra&donec=dallium&npousse&digul&et=
6	Hiller	2078637210	https://economic.com/quis/tupis/egert/maurus/nilia.html?incipit=aloborts+in+kante&turpis&ad&igula&turpis+ne&at&smil&incidunt/au
7	Karola	3011968408	https://paypal.com/aliquam/ugua/quan/sollicitudin/vitae/connetur/egert.png?vel=libero&accumsan=nam&tellus=&dui=isi=prone&le&cor
8	Nada	3447598215	https://youtube.com/nullam/ambit/ilibero/ut.jsp?erat=sapien&verra=sapien&posuere=mon&feugiat=maurus&vitae&ipsum=portitor&porta=

Total rows: 31 of 31 Query complete 00:00:00.194

23. Find clients whose budget is below the average project cost:

Sql:

```
SELECT *
FROM R_Client
WHERE Budget < (SELECT AVG(Cost) FROM R_Project);
```



The screenshot shows the pgAdmin 4 interface with a query editor containing the following SQL code:

```
1 SELECT *
2   FROM R_Client
3 WHERE Budget < (SELECT AVG(Cost) FROM R_Project);
4
```

The results pane displays the following data:

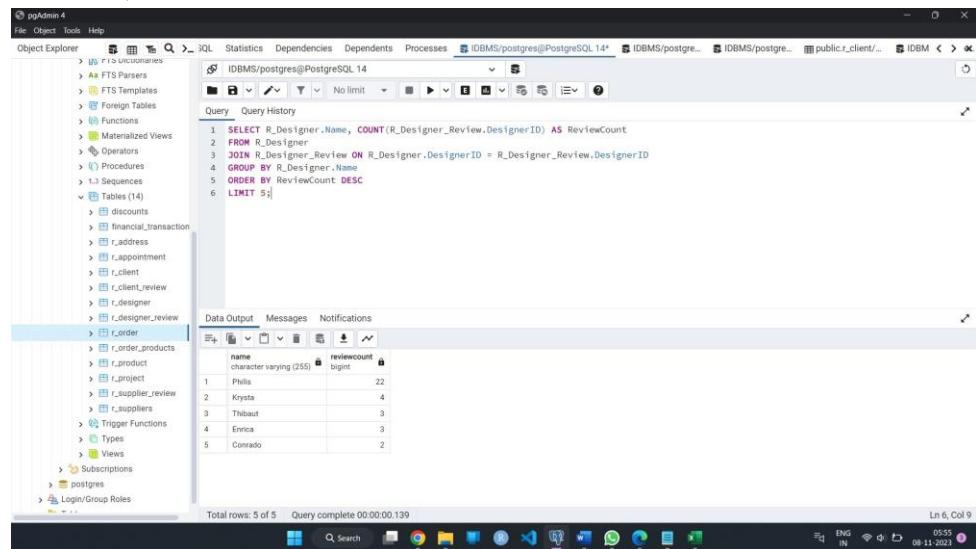
clientid	name	contact	budget	addressid
1	Kushal	9313459333	100.00	4

Total rows: 1 of 1 Query complete 00:00:00.000

24. List the top 5 most reviewed designers:

Sql :

```
SELECT R_Designer.Name, COUNT(R_Designer_Review.DesignerID) AS ReviewCount
FROM R_Designer
JOIN R_Designer_Review ON R_Designer.DesignerID = R_Designer_Review.DesignerID
GROUP BY R_Designer.Name
ORDER BY ReviewCount DESC
LIMIT 5;
```



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The tables listed are: discounts, financial_transaction, r_address, r_appointment, r_client, r_client_review, r_designer, r_designer_review, r_order, r_order_products, r_product, r_project, r_supplier_review, and r_suppliers.
- Query Editor:** Contains the SQL query provided above.
- Data Output:** Displays the results of the query in a table format.
- Table Data:**

	name	reviewcount
1	Phyllis	22
2	Krysta	4
3	Thibaut	3
4	Enrica	3
5	Conrado	2
- Status Bar:** Shows "Total rows: 5 of 5" and "Query complete 00:00:00.139".
- System Tray:** Shows battery level (95%), ENG IN, signal strength, and date/time (08-11-2023).

25. Find all clients who have spent more than the average spend on orders:

Sql:

```
SELECT R_Client.Name, SUM(R_Order.Cost) AS TotalSpent
FROM R_Client
JOIN R_Order ON R_Client.ClientID = R_Order.ClientID
GROUP BY R_Client.Name
HAVING SUM(R_Order.ordercost) > (SELECT AVG(ordercost) FROM R_Order);
```

name	totalspent
Davie	1682476
Annelise	2324320
Nils	2116784
Amil	1747222
Orly	1988701
Aurelia	4873511
Octavius	1627578
Adelbert	2184882

26. Retrieve the average cost of products ordered by each client:

Sql:

```
SELECT R_Client.Name, AVG(R_Product.Cost) AS AverageProductCost
FROM R_Client
JOIN R_Order ON R_Client.ClientID = R_Order.ClientID
JOIN R_Order_Products ON R_Order.OrderID = R_Order_Products.OrderID
JOIN R_Product ON R_Order_Products.ProductID = R_Product.ProductID
GROUP BY R_Client.Name;
```

name	averageproductcost
Theo	11881.000000000000
Eline	10800.000000000000
Pip	7768.330333333333
Davie	6415.500000000000
Elinor	11982.000000000000
Annelise	4852.500000000000
Amil	10491.500000000000
Mariia	2586.000000000000

27. Get a list of products that have never been ordered:

Sql:

```
SELECT *  
FROM R_Product  
WHERE ProductID NOT IN (SELECT ProductID FROM R_Order_Products);
```

The screenshot shows the pgAdmin 4 interface. In the top navigation bar, the connection is listed as 'IDBMS/postgres@PostgreSQL 14*'. The main window has two tabs: 'Query' and 'Query History'. The 'Query' tab contains the following SQL code:

```
-- 12  
SELECT *  
FROM R_Product  
WHERE ProductID NOT IN (SELECT ProductID FROM R_Order_Products);
```

The 'Data Output' tab displays the results of the query. The table structure is:

productid	supplierid	description	cost	quantity
1	259	Orthotrichum flowersii Vitt	12352.00	14
2	276	Camissonia minor (A. Nelson) P.H. Raven	15563.00	9
3	287	Teesdalia coronopifolia (Bergeret) Thell.	21128.00	3
4	215	Dicerandra frutescens Shinners ssp. modesta R.B. Huck	19099.00	15
5	2410	Arenaria fendleri A. Gray	17189.00	4
6	212	Bryum ruticans Brid.	17527.00	18
7	1756	Heuchera americana L. var. americana	21876.00	23
8	2192	Angelica archangelica L.	4384.00	8

Total rows: 20 of 20 Query complete 00:00:00.115

The status bar at the bottom right shows 'Ln 4, Col 65', '05:59', '08-11-2023', and icons for ENG, IN, and battery level.

28. List clients and their last appointment date:

Sql :

```
SELECT R_Client.Name, MAX(R_Appointment.Meeting_Date) AS LastAppointmentDate
FROM R_Client
JOIN R_Appointment ON R_Client.ClientID = R_Appointment.ClientID
GROUP BY R_Client.Name;
```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree view of database objects including tables, functions, and procedures. The central pane contains the SQL query:

```
-- 13
SELECT R_Client.Name, MAX(R_Appointment.Meeting_Date) AS LastAppointmentDate
FROM R_Client
JOIN R_Appointment ON R_Client.ClientID = R_Appointment.ClientID
GROUP BY R_Client.Name;
```

The right pane shows the Data Output tab with the results of the query:

	name	lastappointmentdate
1	Theo	2023-11-03
2	Barnebas	2023-11-03
3	Pip	2023-11-02
4	Maisey	2023-11-09
5	Davie	2023-11-01
6	Elinor	2023-11-09
7	Annelise	2023-11-02
8	Nils	2023-11-04

Total rows: 52 of 52 Query complete 00:00:00.161

29. Show the number of projects per state:

Sql:

```
SELECT R_Address.State, COUNT(R_Project.ProjectID) AS NumberOfProjects  
FROM R_Address  
JOIN R_Client ON R_Address.AddressID = R_Client.AddressID  
JOIN R_Project ON R_Client.ClientID = R_Project.ClientID  
GROUP BY R_Address.State;
```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying a tree view of database objects including tables, functions, and procedures. The central pane contains the SQL query:

```
-- 14  
SELECT R_Address.State, COUNT(R_Project.ProjectID) AS NumberOfProjects  
FROM R_Address  
JOIN R_Client ON R_Address.AddressID = R_Client.AddressID  
JOIN R_Project ON R_Client.ClientID = R_Project.ClientID  
GROUP BY R_Address.State;
```

The right pane shows the Data Output tab, which displays the results of the query:

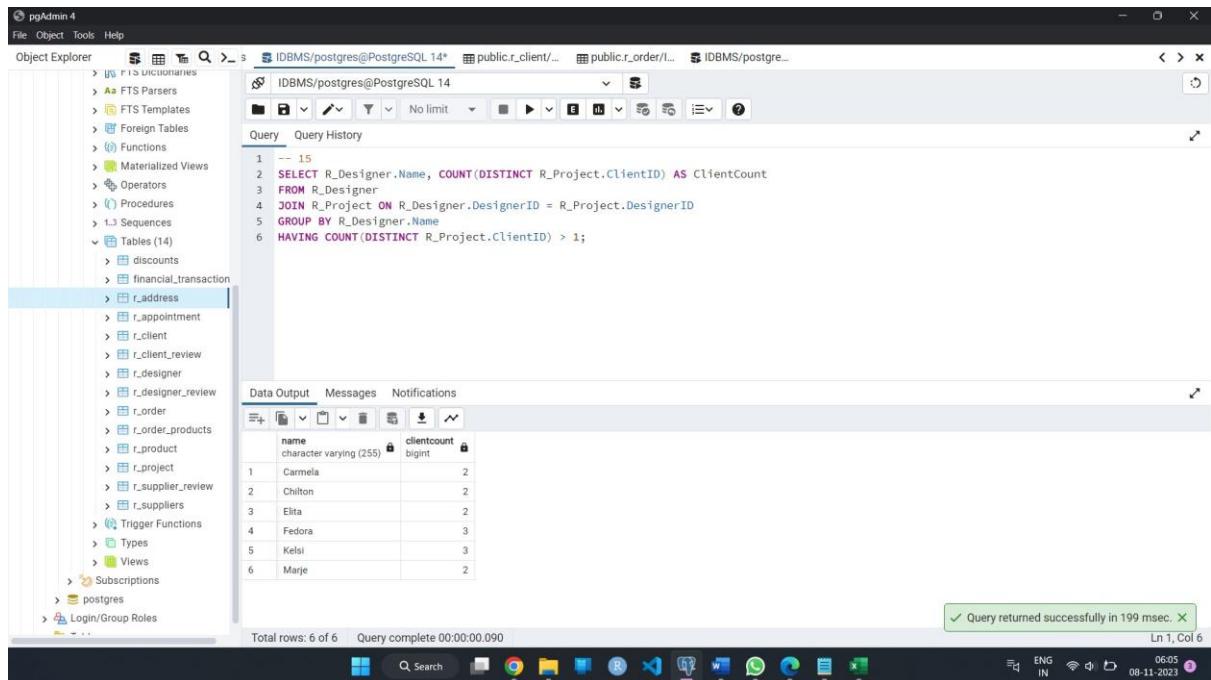
state	numberofprojects
Guanajuato	1
Braga	1
Florida	3
VÄstra Götaland	1
Île-de-France	3
	49

Total rows: 6 of 6 | Query complete 00:00:00.139 | Ln 6, Col 26

30. Find designers who have worked with more than one client:

Sql:

```
SELECT R_Designer.Name, COUNT(DISTINCT R_Project.ClientID) AS ClientCount
FROM R_Designer
JOIN R_Project ON R_Designer.DesignerID = R_Project.DesignerID
GROUP BY R_Designer.Name
HAVING COUNT(DISTINCT R_Project.ClientID) > 1;
```



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database schema with various objects like FTS Parsers, Functions, Materialized Views, Procedures, Sequences, and Tables (14). The table `r.address` is currently selected.
- Query Editor:** Displays the SQL query:

```
-- 15
SELECT R_Designer.Name, COUNT(DISTINCT R_Project.ClientID) AS ClientCount
FROM R_Designer
JOIN R_Project ON R_Designer.DesignerID = R_Project.DesignerID
GROUP BY R_Designer.Name
HAVING COUNT(DISTINCT R_Project.ClientID) > 1;
```

- Data Output:** Shows the results of the query in a table format:

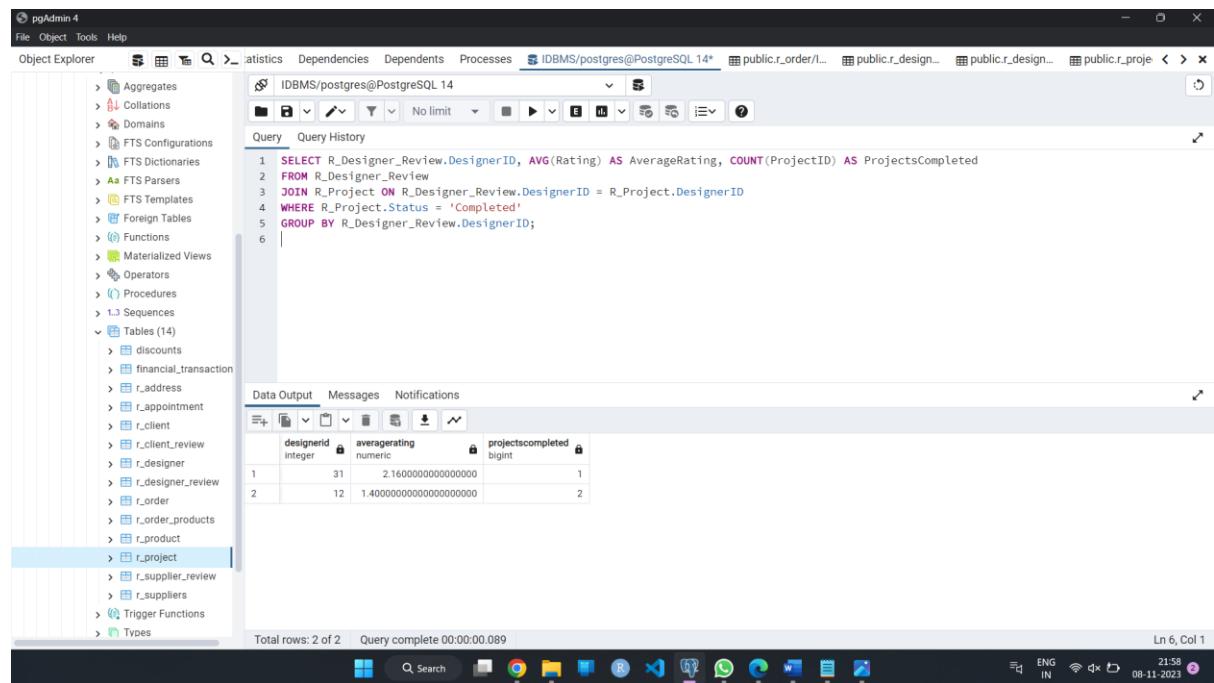
	name	clientcount
1	Carmela	2
2	Chilton	2
3	Elita	2
4	Fedora	3
5	Kelsi	3
6	Marje	2

- Status Bar:** Shows "Query returned successfully in 199 msec." and the timestamp "08-11-2023 06:05".

31. Calculate the average rating of designers by the number of projects they've completed:

Sql:

```
SELECT DesignerID, AVG(Rating) AS AverageRating, COUNT(ProjectID) AS ProjectsCompleted
FROM R_Designer_Review
JOIN R_Project ON R_Designer_Review.DesignerID = R_Project.DesignerID
WHERE R_Project.Status = 'Completed'
GROUP BY DesignerID;
```



```
SELECT R_Designer_Review.DesignerID, AVG(Rating) AS AverageRating, COUNT(ProjectID) AS ProjectsCompleted
FROM R_Designer_Review
JOIN R_Project ON R_Designer_Review.DesignerID = R_Project.DesignerID
WHERE R_Project.Status = 'Completed'
GROUP BY R_Designer_Review.DesignerID;
```

designerid	averagering	projectscompleted
31	2.160000000000000	1
12	1.400000000000000	2

32. Identify the most popular product based on quantity in orders:

Sql:

```
SELECT R_Product.Description, SUM(R_Order_Products.Quantity) AS TotalQuantity
FROM R_Product
JOIN R_Order_Products ON R_Product.ProductID = R_Order_Products.ProductID
GROUP BY R_Product.Description
ORDER BY TotalQuantity DESC
LIMIT 1;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, it lists various database objects under the schema "public". The "r_project" table is currently selected.
- Query Editor:** The main pane contains the SQL query provided in the question. The results show one row:

description	totalquantity
Thelypodium sagittatum (Nutt. ex Torr. & A. Gray) Endl. ex Walp. ssp. ovalifolium (Rydb.) Al-Shethi	84

- Status Bar:** At the bottom, it displays "Total rows: 1 of 1" and "Query complete 00:00:00.162".
- System Icons:** The taskbar at the bottom includes icons for search, file explorer, and other system applications.
- System Information:** The bottom right corner shows the date and time as "08-11-2023 21:59".

33. Show the supplier with the highest average product cost:

Sql:

```
SELECT R_Suppliers.Name, AVG(R_Product.Cost) AS AvgProductCost
FROM R_Suppliers
JOIN R_Product ON R_Suppliers.SupplierID = R_Product.SupplierID
GROUP BY R_Suppliers.Name
ORDER BY AvgProductCost DESC
LIMIT 1;
```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The 'r_suppliers' table is selected. The right pane contains a query editor window with the following SQL code:

```
1 SELECT R_Suppliers.Name, AVG(R_Product.Cost) AS AvgProductCost
2 FROM R_Suppliers
3 JOIN R_Product ON R_Suppliers.SupplierID = R_Product.SupplierID
4 GROUP BY R_Suppliers.Name
5 ORDER BY AvgProductCost DESC
6 LIMIT 1;
```

Below the query editor is a Data Output tab showing the results:

name	avgproductcost
Herrick	21876.000000000000

A status bar at the bottom indicates "Successfully run. Total query runtime: 96 msec. 1 rows affected." and "Ln 7, Col 1". The system tray at the bottom right shows the date and time as 08-11-2023 22:01.

34. List each client's most expensive project:

Sql:

```
SELECT R_Client.Name, R_Project.ProjectID, R_Project.Cost AS  
MostExpensiveProjectCost  
FROM R_Client  
JOIN R_Project ON R_Client.ClientID = R_Project.ClientID  
WHERE R_Project.Cost = (  
    SELECT MAX(Cost)  
    FROM R_Project AS Proj  
    WHERE Proj.ClientID = R_Client.ClientID  
)  
GROUP BY R_Client.Name, R_Project.ProjectID;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, it lists various database objects including Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables (14), and Types. The table `r.project` is currently selected.
- Query Editor:** The main area contains the SQL query provided above. The results of the query are displayed in a table below.
- Data Output:** A table showing the results of the query. The columns are `name`, `projectid`, and `mostexpensiveprojectcost`. The data is as follows:

	name	projectid	mostexpensiveprojectcost
1	Adelbert	58	252.70
2	Agustin	3	591.86
3	Aliza	15	970.47
4	Amil	27	251.21
5	Annelise	1	807.49
6	Arnuad	2	320.53
7	Barnebas	22	204.93
8	Bat	28	711.11

Total rows: 47 of 47 Query complete 00:00:00.162

Successfully run. Total query runtime: 162 msec. 47 rows affected. Ln 10, Col 1

System tray icons at the bottom right include: ENG IN, battery level, signal strength, and date/time (08-11-2023 22:01).

35.A trigger that automatically updates a designer's average rating after a new review is inserted into the R_Designer_Review table:

Sql:

```
CREATE OR REPLACE FUNCTION update_designer_rating()
RETURNS TRIGGER AS $$

BEGIN
    UPDATE R_Designer
    SET Rating = (SELECT AVG(Rating) FROM R_Designer_Review WHERE DesignerID =
NEW.DesignerID)
    WHERE DesignerID = NEW.DesignerID;
    RETURN NEW;
END;

$$ LANGUAGE plpgsql;
CREATE TRIGGER trg_update_designer_rating
AFTER INSERT ON R_Designer_Review
FOR EACH ROW
EXECUTE FUNCTION update_designer_rating();
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects including Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (14). The "r.project" table is currently selected.
- Query Editor:** Contains the SQL code for creating the function and trigger.
- Data Output:** Shows the successful execution of the query with the message: "Query returned successfully in 171 msec."
- System Bar:** Displays the status bar with "Total rows: 47 of 47" and "Query complete 00:00:00.171". It also shows system icons for search, browser, file explorer, and other applications.
- Taskbar:** Shows the Windows taskbar with various pinned icons.

36. A stored procedure to calculate and update the total cost of all orders for a given client:

Sql :

```
CREATE OR REPLACE PROCEDURE update_client_order_total(IN _client_id INT)
LANGUAGE plpgsql
AS $$

DECLARE
    _total DECIMAL(10,2);
BEGIN
    -- Calculate the total cost of all orders for the client
    SELECT INTO _total SUM(R_Order_Products.Quantity * R_Product.Cost)
    FROM R_Order
    INNER JOIN R_Order_Products ON R_Order.OrderID = R_Order_Products.OrderID
    INNER JOIN R_Product ON R_Order_Products.ProductID = R_Product.ProductID
    WHERE R_Order.ClientID = _client_id;
    -- Update the Client's total order cost (assuming there is a column for that)
    UPDATE R_Client
    SET TotalOrderCost = _total
    WHERE ClientID = _client_id;
END;
$$;
```

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, it shows a tree view of database objects. The node for the stored procedure `update_client_order_total` is highlighted.
- Query Editor:** The main window contains the SQL code for the stored procedure. The code is identical to the one provided in the text block above.
- Status Bar:** At the bottom, it displays "Query returned successfully in 85 msec." and "Ln 19, Col 1".
- System Tray:** At the very bottom, it shows system icons for search, browser, file explorer, task manager, and other applications, along with the date and time "08-11-2023" and a battery indicator.

37. A function that retrieves the contact details for a designer given their ID:

Sql:

```
CREATE OR REPLACE FUNCTION get_designer_contact(IN designer_id INT)
RETURNS TABLE(Name VARCHAR, Contact VARCHAR) LANGUAGE plpgsql AS $$  
BEGIN  
RETURN QUERY  
SELECT Name, Contact  
FROM R_Designer  
WHERE DesignerID = designer_id;  
END;  
$$;
```

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a query editor window on the right. The query editor contains the SQL code for creating the function. The status bar at the bottom indicates 'Total rows: 47 of 47' and 'Query complete 00:00:00.064'. The bottom right corner shows the date and time: '08-11-2023 22:14'.

```
CREATE OR REPLACE FUNCTION get_designer_contact(IN designer_id INT)
RETURNS TABLE(Name VARCHAR, Contact VARCHAR) LANGUAGE plpgsql AS $$  
BEGIN  
RETURN QUERY  
SELECT Name, Contact  
FROM R_Designer  
WHERE DesignerID = designer_id;  
END;  
$$;
```

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a query editor window on the right. The query editor contains the SQL code for selecting from the function. The status bar at the bottom indicates 'Total rows: 1 of 1' and 'Query complete 00:00:00.141'. The bottom right corner shows the date and time: '08-11-2023 22:16'.

```
select * from get_designer_contact(3)
```

	name	contact
1	Ashlen	7524522585

38. Find the total number of clients per city and the average budget of clients in that city:

Sql:

```
SELECT R_Address.City, COUNT(DISTINCT R_Client.ClientID) AS NumberOfClients,  
AVG(R_Client.Budget) AS AverageBudget  
FROM R_Client  
JOIN R_Address ON R_Client.AddressID = R_Address.AddressID  
GROUP BY R_Address.City;
```

The screenshot shows the pgAdmin 4 interface with a query editor window. The query is:

```
1 SELECT R_Address.City, COUNT(DISTINCT R_Client.ClientID) AS NumberOfClients, ROUND(AVG(R_Client.Budget),2) AS AverageBudget  
2 FROM R_Client  
3 JOIN R_Address ON R_Client.AddressID = R_Address.AddressID  
4 GROUP BY R_Address.City;  
5
```

The results are displayed in a table:

	city	numberofclients	averagebudget
1	Baishi	1	8804.35
2	Bhimbar	2	7910.59
3	Bulhan	4	5510.90
4	Carhuac	2	7840.03
5	Cheonan	1	3371.54
6	Chilly-Mazarin	5	4781.20
7	Cilated	1	3215.42
8	Clodomira	1	9914.52

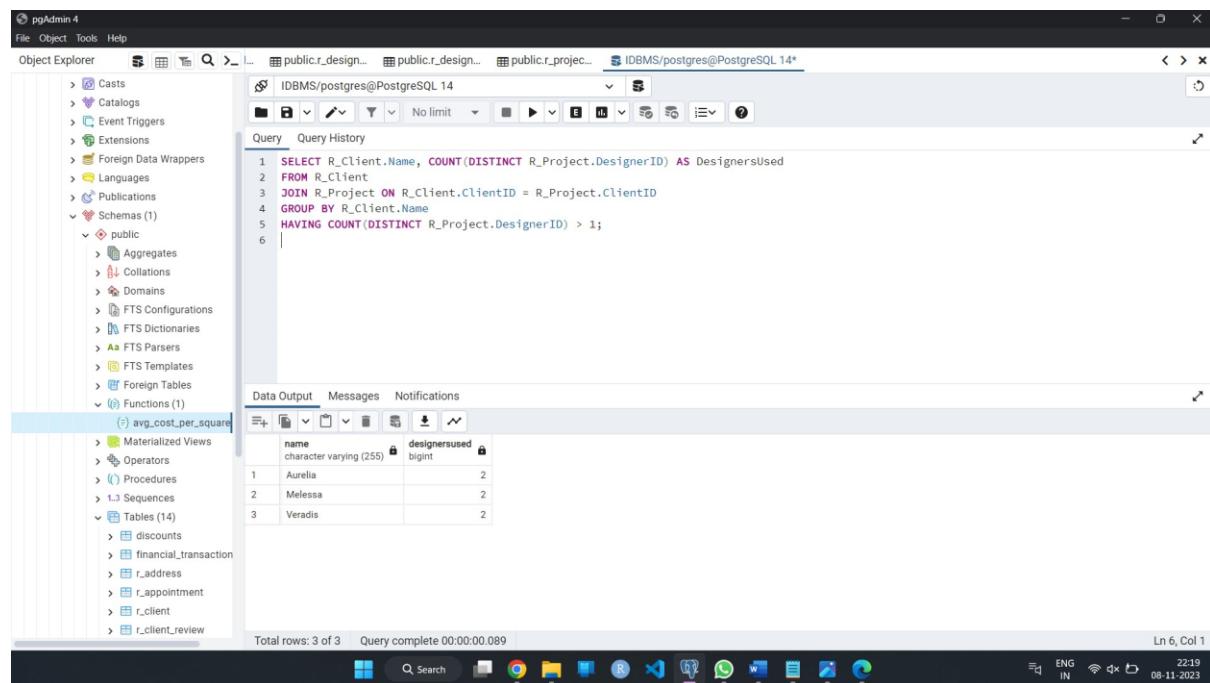
Total rows: 36 of 36 Query complete 00:00:00.083

Message bar: Successfully run. Total query runtime: 83 msec. 36 rows affected.

39. Identify clients who have used multiple designers:

Sql:

```
SELECT R_Client.Name, COUNT(DISTINCT R_Project.DesignerID) AS DesignersUsed
FROM R_Client
JOIN R_Project ON R_Client.ClientID = R_Project.ClientID
GROUP BY R_Client.Name
HAVING COUNT(DISTINCT R_Project.DesignerID) > 1;
```



```
pgAdmin 4
File Object Tools Help
Object Explorer   IDBMS/postgres@PostgreSQL 14*
Query Query History
1 SELECT R_Client.Name, COUNT(DISTINCT R_Project.DesignerID) AS DesignersUsed
2 FROM R_Client
3 JOIN R_Project ON R_Client.ClientID = R_Project.ClientID
4 GROUP BY R_Client.Name
5 HAVING COUNT(DISTINCT R_Project.DesignerID) > 1;
6 |
```

	name	designersused
1	Aurelia	2
2	Melessa	2
3	Veradis	2

Total rows: 3 of 3 Query complete 00:00:00.089 Ln 6, Col 1

Windows Taskbar icons: Search, File Explorer, Task View, Task Manager, Start, Taskbar Buttons, Taskbar Icons.

40. Retrieve the most common appointment agenda topics:

Sql:

```
SELECT Agenda, COUNT(*) AS Frequency
FROM R_Appointment
GROUP BY Agenda
ORDER BY Frequency DESC
LIMIT 1;
```

The screenshot shows the pgAdmin 4 interface. The left pane is the Object Explorer, displaying various database objects like Casts, Catalogs, Event Triggers, Extensions, Foreign Data Wrappers, Languages, Publications, Schemas, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables. The right pane contains a query editor window with the following SQL code:

```
1 SELECT Agenda, COUNT(*) AS Frequency
2 FROM R_Appointment
3 GROUP BY Agenda
4 ORDER BY Frequency DESC
5 LIMIT 1;
6
```

Below the query editor is a Data Output tab showing the results of the query:

agenda	frequency
Papio cynocephalus	2

At the bottom of the pgAdmin window, a status bar indicates "Total rows: 1 of 1" and "Query complete 00:00:00.141". To the right of the status bar, a message box says "Successfully run. Total query runtime: 141 msec. 1 rows affected." The system tray at the bottom right shows the date and time as "08-11-2023 22:20".