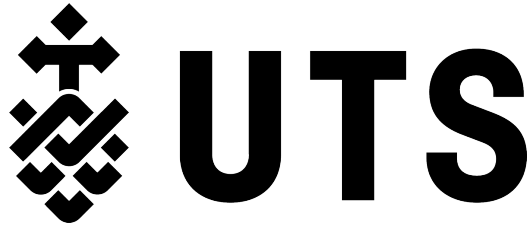


## **ASSESSMENT 2**

### **Analysing Company Performance with SQL**



**94692 Data Science Practice**

**University Of Technology Sydney**

**Lecturer**

Anthony So

**Student name:**

Leah Nguyen

**Date**

October, 2021

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Create and Connect the Database</b>	<b>1</b>
2.1	About The Datasets . . . . .	1
2.2	Create Database . . . . .	1
<b>3</b>	<b>Analysis &amp; Finding</b>	<b>2</b>
3.1	Solution Queries . . . . .	2
3.2	Findings . . . . .	3
3.2.1	The Pricing Strategy . . . . .	3
<b>4</b>	<b>References</b>	<b>4</b>
<b>5</b>	<b>Appendix</b>	<b>5</b>
5.1	Appendix 1: Numer of Data Jobs Mentioning Specific Skills . . . . .	5
5.2	Appendix 2: Entity Relationship Diagram of Northwind datasets . . . . .	6
5.3	Appendix 3: Data Tables Description . . . . .	7
5.4	Appendix 4: Data Dictionaries . . . . .	8
5.5	Appendix 5: Business Questions . . . . .	15
5.6	Appendix 6: Select Query Tool in PGAdmin . . . . .	20
5.7	Appendix 7: Loading datasets into the database . . . . .	21
5.8	Appendix 8: Solution Queries - Question 1 . . . . .	22
5.9	Appendix 9: Solution Queries - Question 2 . . . . .	24
5.10	Appendix 10: Solution Queries - Question 3 . . . . .	25
5.11	Appendix 11: Solution Queries - Question 4 . . . . .	27
5.12	Appendix 12: Solution Queries - Question 5 . . . . .	28
5.13	Appendix 13: Solution Queries - Question 6 . . . . .	30

5.14	Appendix 14: Solution Queries - Question 7 . . . . .	32
5.15	Appendix 15: Solution Queries - Question 8 . . . . .	34
5.16	Appendix 16: Solution Queries - Question 9 . . . . .	40
5.17	Appendix 17: Solution Queries - Question 10 . . . . .	44

# 1 Introduction

Structured Query Language (SQL) is becoming increasingly demanded in companies to evaluate business performance in recent years (Appendix 1). SQL is used for a variety of objectives, including exploratory data analysis, creating a filtered dataset of our data, and even discovering insights about the business itself (Morato, 2020). Knowing how to make data-driven judgments is essential for success in any department, scaling from marketing, finance, product development and so on (Culincu, 2018a). With the given dataset, my goal in this assignment is to use PostgreSQL to analyze the performance and answer important business questions for a fictitious company named Northwind Trader. Additionally, R is also utilized for data visualizations to further explored and produced insights about this dataset.

## 2 Create and Connect the Database

### 2.1 About The Datasets

For big companies, numerous tables are required to store data from multiple sources. An Entity Relationship Diagram - ERD, in this case, can be useful to visualize the relationship between these tables (Culincu, 2018b).

As shown in the EDR for this project (Appendix 2), there are total 13 tables, all connected by at least one field. The data comprises around 2,000+ transactions, including information about customers, employees, payments, products, and orders (Appendix 3, 4).

Through the data has given, different departments of Northwind Traders want to know how well they have performed, as this is more likely to have a positive impact on overall business performance. As the result, there are a total of 10 questions (Appendix 5) designed to uncover these insights, which will be discussed further in the following section of this report. In this assignment, I utilized PGAdmin as the tool to get set-up to do all of these things.

### 2.2 Create Database

The first step is to create a new database and try to import some data from the scripts provided, so we can get started querying it. For that, I used SQL Shell command and execute the following query to create a new database named **DSP – Northwind** :

```
CREATE DATABASE "DSP - Northwind"

WITH

OWNER = postgres

ENCODING = 'UTF8'

CONNECTION LIMIT = -1;
```

Once I have the database, the next step is importing the SQL scripts `northwind.postgre_renamed.sql` and executing the query inside in order to create a new schema called `pubic` and get the tables stored inside the schema. For doing that, I open the **Query Tool** by right click on the Databases (Appendix 6). Inside the query tool, I perform a subsequence of following actions to connect and run the scripts from the local folder (Appendix 7):

1. Chose 'Open File' on the navigation bar.
2. Located the directory to the folder where I have stored the script.
3. Select the `northwind.postgre_renamed.sql` file.
4. Click on 'Select' to import the script.
5. Selecting 'Execute' button on the navigation bar to start loading the datasets into the schema.

As executed the script, PGadmin return the result as following as specifying that the datasets has successfully been retrieved:

GRANT

Query returned successfully in 299 msec.

By understanding the assignment requirements and finishing setting up the database, the next crucial part is to start answers the business questions and visualize data outputs to further explain the insights within departments.

## 3 Analysis & Finding

### 3.1 Solution Queries

In this assignment, there are total 11 questions (Appendix 5) that were requested from 4 departments:

1. The Product Team
2. The Logistics Team
3. The HR Team
4. The Pricing Team

Nevertheless, to have a further understanding of how each of these questions will have an impact on business decisions, I will break down the analysis into 3 sub components, in which each of the component will require some business questions were asked previously:

### **The Pricing Strategy**

- Question 1 (Appendix )
- Question 5
- Question 6
- Question 8

### **Logistics Performance**

- Question 2
- Question 4
- Question 7

### **Sale Performances**

- Question 3
- Question 9
- Question 10

## **3.2 Findings**

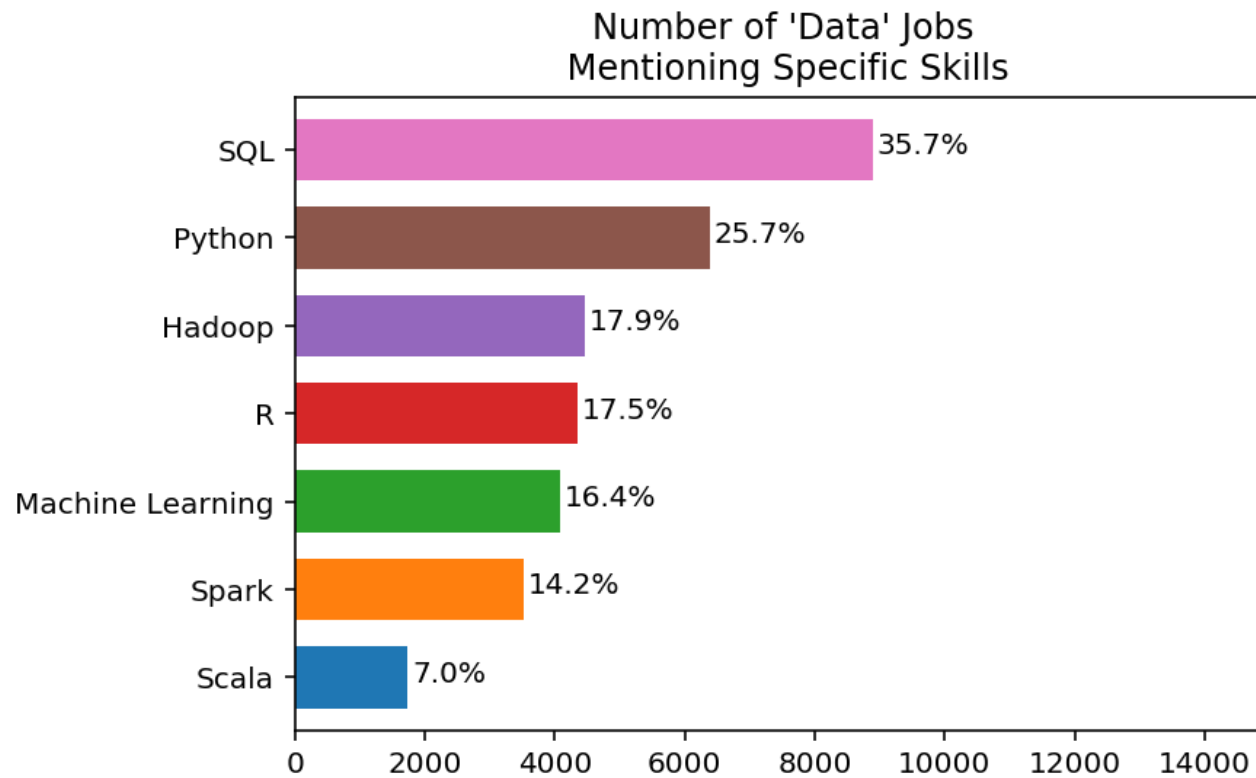
### **3.2.1 The Pricing Strategy**

## 4 References

1. Morato, L. (2020). Answering Business Questions with SQL - Towards Data Science. Medium.  
<https://towardsdatascience.com/answering-business-questions-with-sql-fd0315707980X>
2. Culincu, D. C. (2018a). SQL for Data Analysis — part I - Diana Cristina Culincu. Medium.  
<https://medium.com/@diana.culincu/sql-for-data-analysis-part-i-95ce957b2c4> X
3. Culincu, D. C. (2018b). SQL for Data Analysis — part I - Diana Cristina Culincu. Medium.  
<https://medium.com/@diana.culincu/sql-for-data-analysis-part-i-95ce957b2c4> X
4. Grupman, C. (2021). Want a Job in Data? Learn SQL. Dataquest.  
<https://www.dataquest.io/blog/why-sql-is-the-most-important-language-to-learn/> X

## 5 Appendix

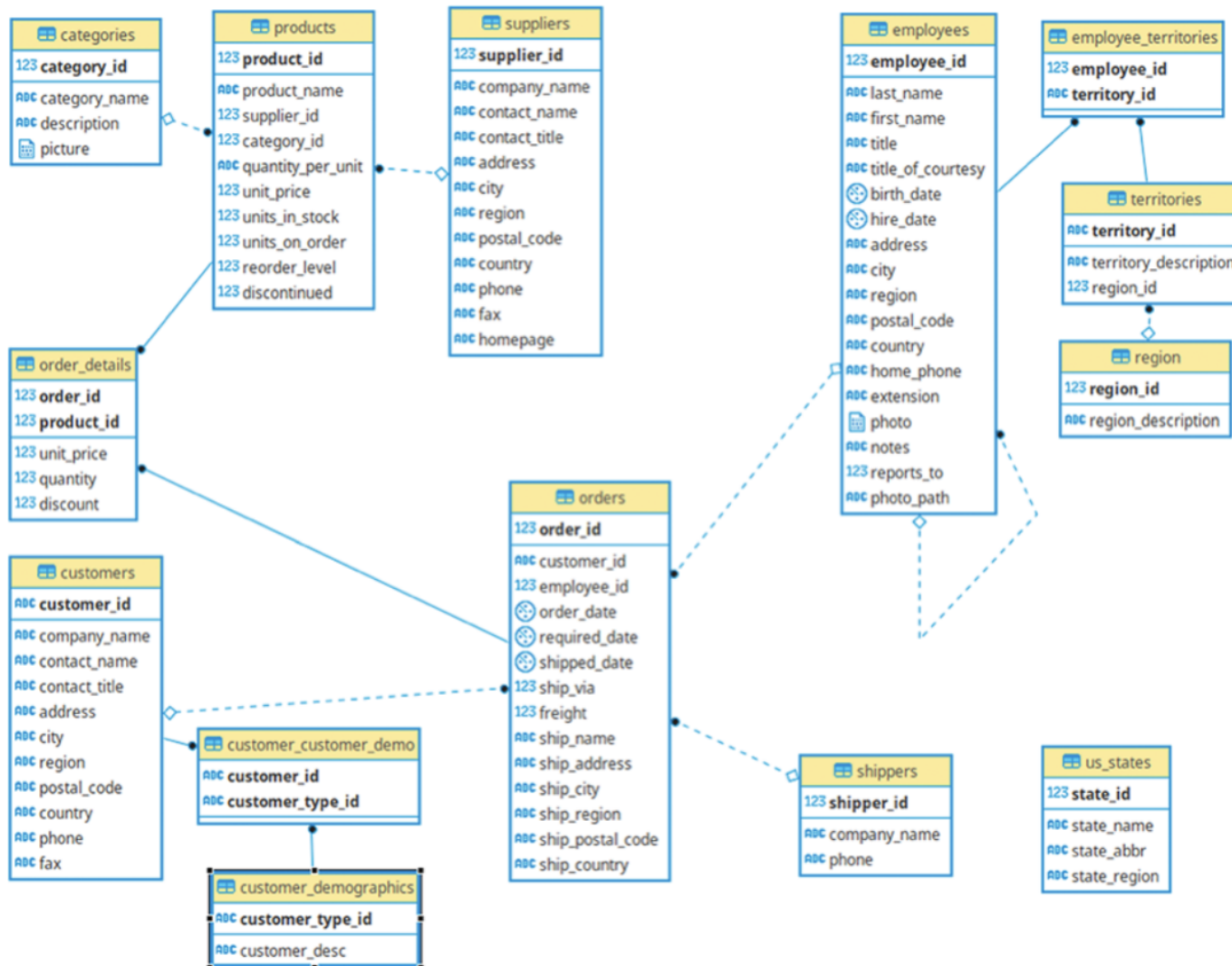
### 5.1 Appendix 1: Numer of Data Jobs Mentioning Specific Skills



Source: Grupman, 2021



## 5.2 Appendix 2: Entity Relationship Diagram of Northwind datasets



### 5.3 Appendix 3: Data Tables Description

Table	Rows
Categories	8
Customers	91
Employees	9
EmployeeTerritories	49
Order Details	2155
Orders	830
Region	4
Products	77
Shippers	6
Suppliers	29
Territories	53
Usstates	51

5.4 Appendix 4: Data Dictionaries

Categories Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	CategoryID	int				Categories of Northwind products.
		IX	CategoryName	varchar	-15			Name of food category.
			Description	text		Yes		Full description of the category. Provide samples.
			Picture	image		Yes		A picture representing the food category.

CustomerCustomerDemo Table

∞

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK	FK	IX	CustomerID	varchar	-5			
PK	FK	IX	CustomerTypeID	varchar	-10			

CustomerDemographics Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	CustomerTypeID	varchar	-10			
			CustomerDesc	ntext		Yes		

Customers Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	CustomerID	varchar	-5			Unique five-character code based on customer name.
		IX	CompanyName	varchar	-40			
			ContactName	varchar	-30	Yes		
			ContactTitle	varchar	-30	Yes		
			Address	varchar	-60	Yes		Street or post-office box.
		IX	City	varchar	-15	Yes		
		IX	Region	varchar	-15	Yes		State or province.
		IX	PostalCode	varchar	-10	Yes		
			Country	varchar	-15	Yes		
			Phone	varchar	-24	Yes		Phone number includes country code or area code.
			Fax	varchar	-24	Yes		Phone number includes country code or area code.

### Employees Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	EmployeeID	int				Number automatically assigned to new employee.
		IX	LastName	varchar	-20			
			FirstName	varchar	-10			

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
			Title	varchar	-30	Yes		Employee's title.
			TitleOfCourtesy	varchar	-25	Yes		
			BirthDate	datetime		Yes		
			HireDate	datetime		Yes		
			Address	varchar	-60	Yes		Street or post-office box.
			City	varchar	-15	Yes		
			Region	varchar	-15	Yes		Street or post-office box.
		IX	PostalCode	varchar	-10	Yes		
			Country	varchar	-15	Yes		
			HomePhone	varchar	-24	Yes		Phone number includes country code or area code.
			Extension	varchar	-4	Yes		Internal telephone extension number.
			Photo	image		Yes		Picture of employee.
			Notes	text		Yes		General information about employee's background.
	FK		ReportsTo	int		Yes		Employee's supervisor.
			PhotoPath	varchar	-255	Yes		

### EmployeeTerritories Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK	FK	IX	EmployeeID	int				
PK	FK	IX	TerritoryID	varchar	-20			

### OrderDetails Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK	FK	IX	OrderID	int				Same as Order ID in Orders table.
PK	FK	IX	ProductID	int				Same as Product ID in Products table.
			UnitPrice	decimal			0	
			Quantity	smallint			-1	
			Discount	decimal			0	

### Orders Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	OrderID	int				
	FK	IX	CustomerID	varchar	-5	Yes		
	FK	IX	EmployeeID	int		Yes		Same entry as in Employees table.
		IX	OrderDate	datetime		Yes		
			RequiredDate	datetime		Yes		
		IX	ShippedDate	datetime		Yes		
	FK	IX	ShipVia	int		Yes		Same as Shipper ID in Shippers table.
			Freight	decimal		Yes	0	
			ShipName	varchar	-40	Yes		Name of person or company to receive the shipment.

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
			ShipAddress	varchar	-60	Yes		Street address only – no post-office box allowed.
			ShipCity	varchar	-15	Yes		
			ShipRegion	varchar	-15	Yes		State or province.
		IX	ShipPostalCode	varchar	-10	Yes		
			ShipCountry	varchar	-15	Yes		

### Products Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	ProductID	int				Number automatically assigned to new product.
		IX	ProductName	nvarchar	-40			
	FK	IX	SupplierID	int		Yes		Same entry as in Suppliers table.
	FK	IX	CategoryID	int		Yes		Same entry as in Categories table.
			QuantityPerUnit	nvarchar	-20	Yes		“(e.g., 24-count case, 1-liter bottle).”
			UnitPrice	money		Yes	0	
			UnitsInStock	smallint		Yes	0	
			UnitsOnOrder	smallint		Yes	0	
			ReorderLevel	smallint		Yes	0	Minimum units to maintain in stock.
			Discontinued	bit			0	Yes means item is no longer available.

Region Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	RegionID	int				
			RegionDescription	varchar	-50			

Shippers Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	ShipperID	int				Number automatically assigned to new shipper.
			CompanyName	varchar	-40			Name of shipping company.
			Phone	varchar	-24	Yes		Phone number includes country code or area code.

Suppliers Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	SupplierID	int				Number automatically assigned to new supplier.
		IX	CompanyName	varchar	-40			
			ContactName	varchar	-30	Yes		
			ContactTitle	varchar	-30	Yes		
			Address	varchar	-60	Yes		Street or post-office box.
			City	varchar	-15	Yes		
			Region	varchar	-15	Yes		State or province.
		IX	PostalCode	varchar	-10	Yes		



Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
			Country	varchar	-15	Yes		
			Phone	varchar	-24	Yes		Phone number includes country code or area code.
			Fax	varchar	-24	Yes		Phone number includes country code or area code.
			HomePage	ntext		Yes		Supplier's home page on World Wide Web.

#### Territories Table

Primary Key	Foreign Key	Index	Name	Data type	Length	Nullable	Default	Description
PK		IX	TerritoryID	varchar	-20			
			TerritoryDescription	varchar	-50			
	FK		RegionID	int				

## 5.5 Appendix 5: Business Questions

**5.5.0.1 Question 1** For their annual review of the company pricing strategy, the Product Team wants to look at the products that are currently being offered for a specific price range (\$20 to \$50). In order to help them they asked you to provide them with a list of products with the following information:

1. their name
2. their unit price

Filtered on the following conditions:

1. their unit price is between 20 and 50
2. they are not discontinued

Finally order the results by unit price in a descending order (highest first).

**5.5.0.2 Question 2** The Logistics Team wants to do a retrospection of their performances for the year 1998, in order to identify for which countries they didn't perform well. They asked you to provide them a list of countries with the following information:

1. their average days between the order date and the shipping date (formatted to have only 2 decimals)
2. their total number of unique orders (based on the order id)

Filtered on the following conditions:

1. the year of order date is 1998
2. their average days between the order date and the shipping date is greater or equal 5 days
3. their total number of orders is greater than 10 orders

Finally order the results by country name in an ascending order (following alphabetical order).

**5.5.0.3 Question 3** The HR Team wants to know for each employee what was their age on the date they joined the company and who they currently report to. Provide them with a list of every employees with the following information:

1. their full name (first name and last name combined in a single field)
2. their job title
3. their age at the time they were hired
4. their manager full name (first name and last name combined in a single field)
5. their manager job title

Finally order the results by employee age and employee full name in an ascending order (lowest first).

**5.5.0.4 Question 4** The Logistics Team wants to do a retrospection of their global performances over 1997-1998, in order to identify for which month they perform well. They asked you to provide them a list with:

1. their year/month as single field in a date format (e.g. "1996-01-01" January 1996)
2. their total number of orders
3. their total freight (formatted to have no decimals)

Filtered on the following conditions:

1. the order date is between 1997 and 1998
2. their total number of orders is greater than 35 orders

Finally order the results by total freight (descending order).

**5.5.0.5 Question 5** The Pricing Team wants to know which products had an unit price increase and the percentage increase was not between 20% and 30%. In order to help them they asked you to provide them a list of products with:

1. their product name
2. their current unit price (formatted to have only 2 decimals)
3. their initial unit price (formatted to have only 2 decimals)
4. their percentage increase as:

$(\text{Current Unit Price} - \text{Initial Unit Price}) \div \text{Initial Unit Price} \times 100$  (with the result formatted to an integer, e.g 50 for 50%)

Filtered on the following conditions:

1. their percentage increase is not between 20% and 30%
2. their total number of orders is greater than 10 orders

Finally order the results by percentage increase (ascending order).

**5.5.0.6 Question 6** The Pricing Team wants to know how each category performs according to their price range. In order to help them they asked you to provide them a list of categories with:

1. their category name
2. their price range as:
3. "1.Below \$20"
4. "2. \$20 - \$50"
5. "3. Over \$50"
6. their total amount (formatted to be integer) taking into account the offered discount (subtracting the discounted amount)
7. their volume of orders (number of orders in which the category was present)

Finally order the results by category name then price range (both ascending order).

**5.5.0.7 Question 7** The Logistics Team wants to know what is the current state of our regional suppliers' stocks for each category of product. In order to help them they asked you to provide them a list of categories with:

1. their supplier region" as:
2. "America"
3. "Europe"
4. "Asia-Pacific"
5. their category name
6. their total units in stock
7. their total units on order

8. their total reorder level

Finally order the results by category name, then supplier region and reorder level (each in ascending order).

**5.5.0.8 Question 8** The Pricing Team wants to know for each currently offered product how their unit price compares against their categories average and median unit price. In order to help them they asked you to provide them a list of products with:

1. their category name
2. their product name
3. their unit price
4. their category average unit price (formatted to have only 2 decimals)
5. their category median unit price (formatted to have only 2 decimals)
6. their position against the category average unit price as:
7. "Below Average"
8. "Equal Average"
9. "Over Average"
10. their position against the category median unit price as:
11. "Below Median"
12. "Equal Median"
13. "Over Median"

Filtered on the following conditions:

1. They are not discontinued

Finally order the results by category name then product name (both ascending).

**5.5.0.9 Question 9** The Sales Team wants to build a list of KPIs to measure employees' performances. In order to help them they asked you to provide them a list of employees with:

1. their full name (first name and last name combined in a single field)
2. their job title
3. their total sales amount excluding discount (formatted to have only 2 decimals)

4. their total number of unique orders
5. their total number of orders
6. their average product amount excluding discount (formatted to have only 2 decimals)
7. their average order amount excluding discount (formatted to have only 2 decimals)
8. their total discount amount (formatted to have only 2 decimals)
9. their total sales amount including discount (formatted to have only 2 decimals)
10. Their total discount percentage (formatted to have only 2 decimals)

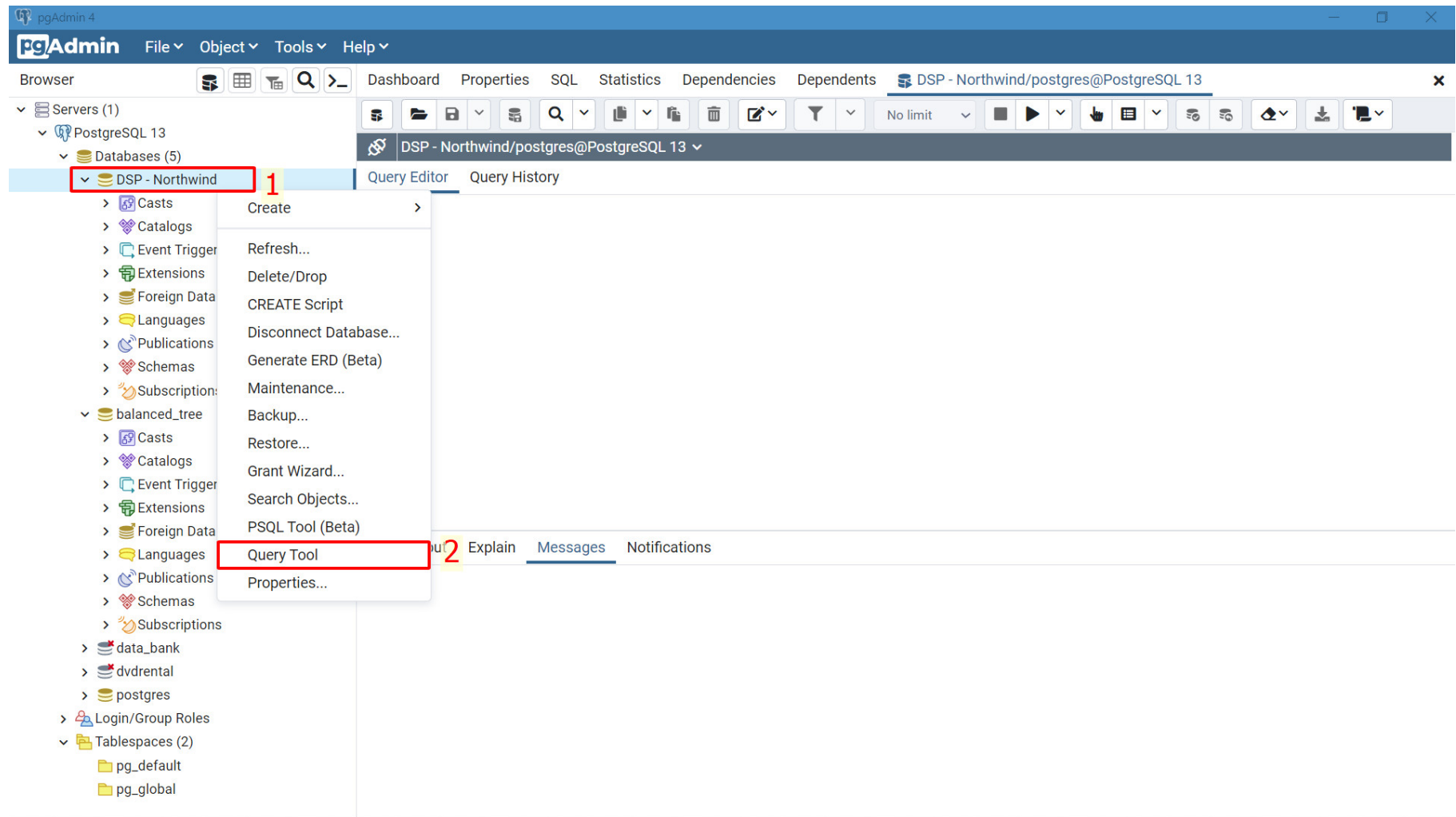
Finally order the results by total sales amount including discount (descending).

**5.5.0.10 Question 10** The Sales Team wants to build another list of KPIs to measure employees' performances across each category. In order to help them they asked you to provide them a list of categories and employees with:

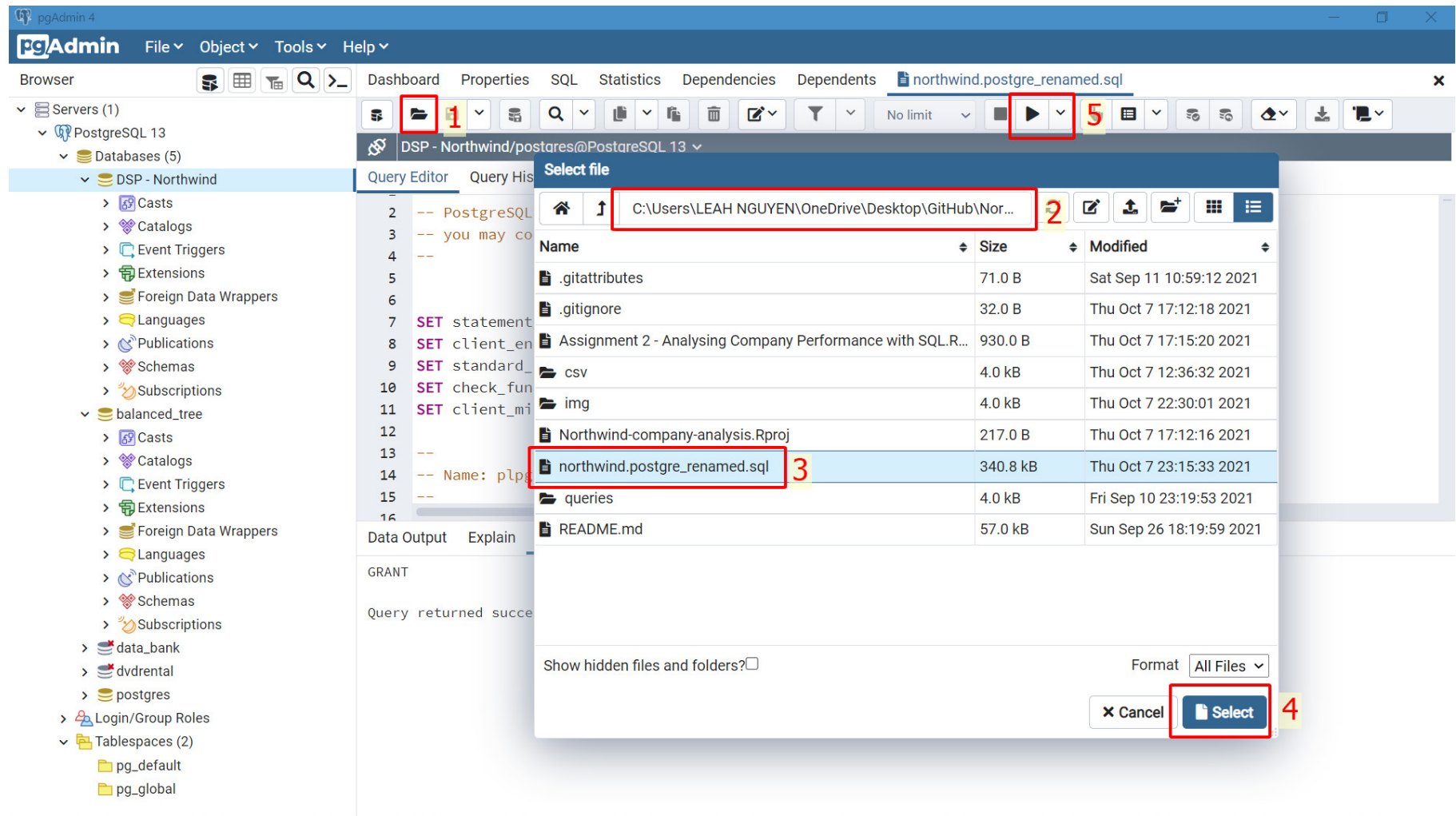
1. their categories name
2. their full name (first name and last name combined in a single field)
3. their total sales amount including discount (formatted to have only 2 decimals)
4. their percentage of total sales amount including discount against the total sales amount across all employees (formatted to have only 2 decimals)
5. their percentage of total sales amount including discount against the total sales amount for each employees (formatted to have only 2 decimals)

Finally order the results by category name (ascending) then total sales amount (descending).

## 5.6 Appendix 6: Select Query Tool in PGadmin



## 5.7 Appendix 7: Loading datasets into the database





## 5.8 Appendix 8: Solution Queries - Question 1

```
-- Solution Query
SELECT
    product_name,
    unit_price
FROM products
WHERE unit_price BETWEEN 20 AND 50
AND discontinued = 0
ORDER BY unit_price DESC;
```

“product_name”	“unit_price”
“Tarte au sucre”	49.3
“Ipoh Coffee”	46
“Vegie-spread”	43.9
“Schoggi Schokolade”	43.9
“Northwoods Cranberry Sauce”	40
“Gnocchi di nonna Alice”	38
“Queso Manchego La Pastora”	38
“Gudbrandsdalsost”	36
“Mozzarella di Giovanni”	34.8
“Camembert Pierrot”	34
“Wimmers gute Semmelknödel”	33.25
“Mascarpone Fabioli”	32
“Gumbär Gummibärchen”	31.23
“Ikura”	31
“Uncle Bob’s Organic Dried Pears”	30
“Sirop d’érable”	28.5
“Gravad lax”	26
“Nord-Ost Matjeshering”	25.89
“Grandma’s Boysenberry Spread”	25
“Pâté chinois”	24
“Tofu”	23.25

“product_name”	“unit_price”
“Chef Anton’s Cajun Seasoning”	22
“Flotemysost”	21.5
“Louisiana Fiery Hot Pepper Sauce”	21.05
“Queso Cabrales”	21
“Gustaf’s Knäckebröd”	21
“Maxilaku”	20

## 5.9 Appendix 9: Solution Queries - Question 2

```
-- Solution Query
WITH cte_avg_days AS (
  SELECT
    ship_country,
    ROUND(AVG(
      EXTRACT(DAY FROM (shipped_date - order_date) * INTERVAL '1 DAY')
    )::NUMERIC,
    2) AS average_days_between_order_shipping,
    COUNT(*) AS total_number_orders
  FROM orders
  WHERE EXTRACT(YEAR FROM order_date) = 1998
  GROUP BY
    ship_country
  ORDER BY ship_country
)
SELECT * FROM cte_avg_days
WHERE average_days_between_order_shipping >= 5
AND total_number_orders > 10;
```

"ship_country"	"average_days_between_order_shipping"	"total_number_orders"
"Austria"	"5.89"	"11"
"Brazil"	"8.12"	"28"
"France"	"9.43"	"23"
"Germany"	"5.38"	"34"
"Spain"	"7.83"	"12"
"Sweden"	"13.29"	"14"
"UK"	"6.25"	"16"
"USA"	"7.89"	"39"
"Venezuela"	"8.73"	"18"

## 5.10 Appendix 10: Solution Queries - Question 3

-- Solution Query

SELECT

```
    CONCAT(e.first_name, ' ', e.last_name) AS employee_full_name,
    e.title AS employee_title,
    EXTRACT(YEAR FROM AGE(e.hire_date, e.birth_date))::INT AS employee_age,
    CONCAT(m.first_name, ' ', m.last_name) AS manager_full_name,
    m.title AS manager_title
```

FROM

```
    employees AS e
```

INNER JOIN employees AS m

ON m.employee\_id = e.reports\_to

ORDER BY

```
    employee_age,
    employee_full_name;
```

"employee_full_name" "employee_title"		"employee_age"	"manager_full_name" "manager_title"	
"Anne Dodsworth"	"Sales Representative"	28	"Steven Buchanan"	"Sales Manager"
"Janet Leverling"	"Sales Representative"	28	"Andrew Fuller"	"Vice President, Sales"
"Michael Suyama"	"Sales Representative"	30	"Steven Buchanan"	"Sales Manager"
"Robert King"	"Sales Representative"	33	"Steven Buchanan"	"Sales Manager"
"Laura Callahan"	"Inside Sales Coordinator"	36	"Andrew Fuller"	"Vice President, Sales"
"Steven Buchanan"	"Sales Manager"	38	"Andrew Fuller"	"Vice President, Sales"
"Nancy Davolio"	"Sales Representative"	43	"Andrew Fuller"	"Vice President, Sales"

“employee_full_name”“employee_title”		“employee_age”	“manager_full_name”	“manager_title”
“Margaret Peacock”	“Sales Representative”	55	“Andrew Fuller”	“Vice President, Sales”

## 5.11 Appendix 11: Solution Queries - Question 4

```
-- Solution Query
WITH cte_freight AS (
  SELECT
    CONCAT(EXTRACT(YEAR FROM order_date),
           '-',
           EXTRACT(MONTH FROM order_date),
           '-01'
          ) AS year_month,
    COUNT(*) AS total_number_orders,
    ROUND(
      SUM(freight)
    )::INT AS total_freight
  FROM orders
  WHERE order_date >= '1997-01-01' AND order_date < '1998-01-01'
  GROUP BY
    CONCAT(EXTRACT(YEAR FROM order_date),
           '-',
           EXTRACT(MONTH FROM order_date),
           '-01'
          )
)
SELECT * FROM cte_freight
WHERE total_number_orders > 35
ORDER BY total_freight DESC;
```

"year_month"	"total_number_orders"	"total_freight"
"1997-10-01"	38	3946
"1997-12-01"	48	3758
"1997-9-01"	37	3237

## 5.12 Appendix 12: Solution Queries - Question 5

```
-- Solution Query
WITH cte_price AS (
SELECT
    d.product_id,
    p.product_name,
    ROUND(LEAD(d.unit_price) OVER (PARTITION BY p.product_name ORDER BY o.order_date)::NUMERIC,2) AS current_price,
    ROUND(LAG(d.unit_price) OVER (PARTITION BY p.product_name ORDER BY o.order_date)::NUMERIC,2) AS previous_unit_price
FROM products AS p
INNER JOIN order_details AS d
ON p.product_id = d.product_id
INNER JOIN orders AS o
ON d.order_id = o.order_id
)
SELECT
    c.product_name,
    c.current_price,
    c.previous_unit_price,
    ROUND(100*(c.current_price - c.previous_unit_price)/c.previous_unit_price) AS percentage_increase
FROM cte_price AS c
INNER JOIN order_details AS d
ON c.product_id = d.product_id
WHERE c.current_price != c.previous_unit_price
```

```

GROUP BY
    c.product_name,
    c.current_price,
    c.previous_unit_price
HAVING COUNT(DISTINCT d.order_id) > 10
AND ROUND(100*(c.current_price - c.previous_unit_price)/c.previous_unit_price) NOT BETWEEN 20 AND 30;

```

“product_name”	“current_price”	“previous_unit_price”	“percentage_increase”
“Mozzarella di Giovanni”	27.8	34.8	-20
“Singaporean Hokkien Fried Mee”	11.2	9.8	14



### 5.13 Appendix 13: Solution Queries - Question 6

```
--Solution query
SELECT
    c.category_name,
    CASE
        WHEN p.unit_price < 20 THEN '1. Below $20'
        WHEN p.unit_price >= 20 AND p.unit_price <= 50 THEN '2. $20 - $50'
        WHEN p.unit_price > 50 THEN '3. Over $50'
    END AS price_range,
    ROUND(SUM(d.unit_price * d.quantity)) AS total_amount,
    COUNT(DISTINCT d.order_id) AS total_number_orders
FROM categories AS c
INNER JOIN products AS p
ON c.category_id = p.category_id
INNER JOIN order_details AS d
ON d.product_id = p.product_id
GROUP BY
    c.category_name,
    price_range
ORDER BY
    c.category_name,
    price_range;
```

"category_name"	"price_range"	"total_amount"	"total_number_orders"
"Beverages"	"1. Below \$20"	111464	317
"Beverages"	"2. \$20 - \$50"	25079	28
"Beverages"	"3. Over \$50"	149984	24
"Condiments"	"1. Below \$20"	28622	85
"Condiments"	"2. \$20 - \$50"	85073	121
"Confections"	"1. Below \$20"	57369	197
"Confections"	"2. \$20 - \$50"	96094	106
"Confections"	"3. Over \$50"	23636	16

“category_name”	“price_range”	“total_amount”	“total_number_orders”
“Dairy Products”	“1. Below \$20”	17886	81
“Dairy Products”	“2. \$20 - \$50”	157148	204
“Dairy Products”	“3. Over \$50”	76296	54
“Grains/Cereals”	“1. Below \$20”	25364	99
“Grains/Cereals”	“2. \$20 - \$50”	75363	91
“Meat/Poultry”	“1. Below \$20”	5121	36
“Meat/Poultry”	“2. \$20 - \$50”	76504	96
“Meat/Poultry”	“3. Over \$50”	96563	36
“Produce”	“1. Below \$20”	2566	13
“Produce”	“2. \$20 - \$50”	57960	81
“Produce”	“3. Over \$50”	44743	39
“Seafood”	“1. Below \$20”	69673	217
“Seafood”	“2. \$20 - \$50”	39963	70
“Seafood”	“3. Over \$50”	31988	27

## 5.14 Appendix 14: Solution Queries - Question 7

```
--Solution query
SELECT
    c.category_name,
    CASE
        WHEN s.country IN ('Australia', 'Singapore', 'Japan' ) THEN 'Asia-Pacific'
        WHEN s.country IN ('US', 'Brazil', 'Canada') THEN 'America'
        ELSE 'Europe'
    END AS supplier_region,
    p.unit_in_stock AS units_in_stock,
    p.unit_on_order AS units_on_order,
    p.reorder_level
FROM suppliers AS s
INNER JOIN products AS p
ON s.supplier_id = p.supplier_id
INNER JOIN categories AS c
ON p.category_id = c.category_id
WHERE s.region IS NOT NULL
ORDER BY
    supplier_region,
    c.category_name,
    p.unit_price;
```

“category_name”	“supplier_region”	“units_in_stock”	“units_on_order”	“reorder_level”
“Condiments”	“America”	113	0	25
“Confections”	“America”	17	0	0
“Meat/Poultry”	“America”	21	0	10
“Meat/Poultry”	“America”	115	0	20
“Beverages”	“Asia-Pacific”	15	10	30
“Condiments”	“Asia-Pacific”	24	0	5
“Confections”	“Asia-Pacific”	29	0	10
“Grains/Cereals”	“Asia-Pacific”	38	0	25

“category_name”	“supplier_region”	“units_in_stock”	“units_on_order”	“reorder_level”
“Meat/Poultry”	“Asia-Pacific”	0	0	0
“Meat/Poultry”	“Asia-Pacific”	0	0	0
“Produce”	“Asia-Pacific”	20	0	10
“Seafood”	“Asia-Pacific”	42	0	0
“Beverages”	“Europe”	52	0	10
“Beverages”	“Europe”	111	0	15
“Beverages”	“Europe”	20	0	15
“Condiments”	“Europe”	4	100	20
“Condiments”	“Europe”	76	0	0
“Condiments”	“Europe”	0	0	0
“Condiments”	“Europe”	53	0	0
“Condiments”	“Europe”	120	0	25
“Condiments”	“Europe”	6	0	0
“Dairy Products”	“Europe”	22	30	30
“Dairy Products”	“Europe”	86	0	0
“Produce”	“Europe”	15	0	10
“Seafood”	“Europe”	85	0	10
“Seafood”	“Europe”	123	0	30

## 5.15 Appendix 15: Solution Queries - Question 8

```
--Solution query
WITH cte_price AS (
    SELECT
        c.category_name,
        p.product_name,
        p.unit_price,
        ROUND(AVG(d.unit_price)::NUMERIC,2) AS average_unit_price,
        ROUND((PERCENTILE_CONT(0.5) WITHIN GROUP (ORDER BY d.unit_price))::NUMERIC,
            2) AS median_unit_price
    FROM categories AS c
    INNER JOIN products AS p
    ON c.category_id = p.category_id
    INNER JOIN order_details AS d
    ON p.product_id = d.product_id
    WHERE p.discontinued = 0
    GROUP BY
        c.category_name,
        p.product_name,
        p.unit_price
)
SELECT
    category_name,
```

```

product_name,
unit_price,
average_unit_price,
median_unit_price,
CASE
    WHEN unit_price > average_unit_price THEN 'Over Average'
    WHEN unit_price = average_unit_price THEN 'Equal Average'
    WHEN unit_price < average_unit_price THEN 'Below Average'
END AS average_unit_price_position,
CASE
    WHEN unit_price > median_unit_price THEN 'Over Average'
    WHEN unit_price = median_unit_price THEN 'Equal Average'
    WHEN unit_price < median_unit_price THEN 'Below Average'
END AS median_unit_price_position
FROM cte_price
ORDER BY
    category_name,
    product_name;

```

"category_name"	"product_name"	"unit_price"	"average_unit_price"	"median_unit_price"	"average_unit_price_position"	"median_unit_price_position"
"Beverages"	"Côte de Blaye"	263.5	245.93	263.50	"Over Average"	"Equal Average"
"Beverages"	"Chartreuse verte"	18	16.68	18.00	"Over Average"	"Equal Average"
"Beverages"	"Ipoh Coffee"	46	43.04	46.00	"Over Average"	"Equal Average"

“category_name”	“product_name”	“unit_price”	“average_unit_price”	“median_unit_price”	“average_unit_price_position”	“median_unit_price_position”
“Beverages”	“Lakkalikööri”	18	16.98	18.00	“Over Average”	“Equal Average”
“Beverages”	“Laughing Lumberjack Lager”	14	13.72	14.00	“Over Average”	“Equal Average”
“Beverages”	“Outback Lager”	15	14.15	15.00	“Over Average”	“Equal Average”
“Beverages”	“Rhönbräu Klosterbier”	7.75	7.38	7.75	“Over Average”	“Equal Average”
“Beverages”	“Sasquatch Ale”	14	12.97	14.00	“Over Average”	“Equal Average”
“Beverages”	“Steeleye Stout”	18	17.00	18.00	“Over Average”	“Equal Average”
“Condiments”	“Aniseed Syrup”	10	9.50	10.00	“Over Average”	“Equal Average”
“Condiments”	“Chef Anton’s Cajun Seasoning”	22	20.68	22.00	“Over Average”	“Equal Average”
“Condiments”	“Genen Shouyu”	13	14.47	15.50	“Below Average”	“Below Average”
“Condiments”	“Grandma’s Boysenberry Spread”	25	24.17	25.00	“Over Average”	“Equal Average”
“Condiments”	“Gula Malacca”	19.45	18.13	19.45	“Over Average”	“Over Average”
“Condiments”	“Louisiana Fiery Hot Pepper Sauce”	21.05	19.46	21.05	“Over Average”	“Below Average”
“Condiments”	“Louisiana Hot Spiced Okra”	17	15.30	15.30	“Over Average”	“Over Average”
“Condiments”	“Northwoods Cranberry Sauce”	40	38.77	40.00	“Over Average”	“Equal Average”
“Condiments”	“Original Frankfurter grüne Soße”	13	12.11	13.00	“Over Average”	“Equal Average”

“category_name”	“product_name”	“unit_price”	“average_unit_price”	“median_unit_price”	“average_unit_price_position”	“median_unit_price_position”
“Condiments”	“Sirop d’érable”	28.5	27.79	28.50	“Over Average”	“Equal Average”
“Condiments”	“Vegie-spread”	43.9	40.79	43.90	“Over Average”	“Over Average”
“Confections”	“Chocolade”	12.75	11.90	12.75	“Over Average”	“Equal Average”
“Confections”	“Gumbär Gummibärchen”	31.23	28.86	31.23	“Over Average”	“Below Average”
“Confections”	“Maxilaku”	20	18.48	20.00	“Over Average”	“Equal Average”
“Confections”	“NuNuCa Nuß-Nougat-Creme”	14	13.07	14.00	“Over Average”	“Equal Average”
“Confections”	“Pavlova”	17.45	16.38	17.45	“Over Average”	“Over Average”
“Confections”	“Scottish Longbreads”	12.5	11.54	12.50	“Over Average”	“Equal Average”
“Confections”	“Schoggi Schokolade”	43.9	40.97	43.90	“Over Average”	“Over Average”
“Confections”	“Sir Rodney’s Marmalade”	81	75.94	81.00	“Over Average”	“Equal Average”
“Confections”	“Sir Rodney’s Scones”	10	9.38	10.00	“Over Average”	“Equal Average”
“Confections”	“Tarte au sucre”	49.3	46.41	49.30	“Over Average”	“Below Average”
“Confections”	“Teatime Chocolate Biscuits”	9.2	8.53	9.20	“Over Average”	“Below Average”
“Confections”	“Valkoinen suklaa”	16.25	14.95	16.25	“Over Average”	“Equal Average”
“Confections”	“Zaanse koeken”	9.5	9.14	9.50	“Over Average”	“Equal Average”
“Dairy Products”	“Camembert Pierrot”	34	32.13	34.00	“Over Average”	“Equal Average”
“Dairy Products”	“Flotemysost”	21.5	19.76	21.50	“Over Average”	“Equal Average”



“category_name”	“product_name”	“unit_price”	“average_unit_price”	“median_unit_price”	“average_unit_price_position”	“median_unit_price_position”
“Dairy Products”	“Geitost”	2.5	2.33	2.50	“Over Average”	“Equal Average”
“Dairy Products”	“Gorgonzola Telino”	12.5	11.67	12.50	“Over Average”	“Equal Average”
“Dairy Products”	“Gudbrandsdalsost”	36	33.45	36.00	“Over Average”	“Equal Average”
“Dairy Products”	“Mascarpone Fabioli”	32	30.72	32.00	“Over Average”	“Equal Average”
“Dairy Products”	“Mozzarella di Giovanni”	34.8	32.04	34.80	“Over Average”	“Below Average”
“Dairy Products”	“Queso Cabrales”	21	19.60	21.00	“Over Average”	“Equal Average”
“Dairy Products”	“Queso Manchego La Pastora”	38	36.91	38.00	“Over Average”	“Equal Average”
“Dairy Products”	“Raclette Courdavault”	55	51.13	55.00	“Over Average”	“Equal Average”
“Grains/Cereals”	“Filo Mix”	7	6.76	7.00	“Over Average”	“Equal Average”
“Grains/Cereals”	“Gnocchi di nonna Alice”	38	35.42	38.00	“Over Average”	“Equal Average”
“Grains/Cereals”	“Gustaf’s Knäckebröd”	21	20.40	21.00	“Over Average”	“Equal Average”
“Grains/Cereals”	“Ravioli Angelo”	19.5	18.14	19.50	“Over Average”	“Equal Average”
“Grains/Cereals”	“Tunnbröd”	9	8.37	9.00	“Over Average”	“Equal Average”

“category_name”	“product_name”	“unit_price”	“average_unit_price”	“median_unit_price”	“average_unit_price_position”	“median_unit_price_position”
“Grains/Cereals”	“Wimmers gute Semmelknödel”	33.25	31.03	33.25	“Over Average”	“Equal Average”
“Meat/Poultry”	“Pâté chinois”	24	22.40	24.00	“Over Average”	“Equal Average”
“Meat/Poultry”	“Tourtière”	7.45	6.80	7.45	“Over Average”	“Below Average”
“Produce”	“Longlife Tofu”	10	8.77	8.00	“Over Average”	“Over Average”
“Produce”	“Manjimup Dried Apples”	53	50.55	53.00	“Over Average”	“Equal Average”
“Produce”	“Tofu”	23.25	21.35	23.25	“Over Average”	“Equal Average”
“Produce”	“Uncle Bob’s Organic Dried Pears”	30	29.17	30.00	“Over Average”	“Equal Average”
“Seafood”	“Boston Crab Meat”	18.4	17.23	18.40	“Over Average”	“Below Average”
“Seafood”	“Carnarvon Tigers”	62.5	59.72	62.50	“Over Average”	“Equal Average”
“Seafood”	“Escargots de Bourgogne”	13.25	12.66	13.25	“Over Average”	“Equal Average”
“Seafood”	“Gravad lax”	26	23.40	23.40	“Over Average”	“Over Average”
“Seafood”	“Ikura”	31	29.68	31.00	“Over Average”	“Equal Average”
“Seafood”	“Inlagd Sill”	19	17.90	19.00	“Over Average”	“Equal Average”
“Seafood”	“Jack’s New England Clam Chowder”	9.65	9.19	9.65	“Over Average”	“Below Average”
“Seafood”	“Konbu”	6	5.76	6.00	“Over Average”	“Equal Average”
“Seafood”	“Nord-Ost Matjeshering”	25.89	24.27	25.89	“Over Average”	“Below Average”
“Seafood”	“Röd Kaviar”	15	14.36	15.00	“Over Average”	“Equal Average”
“Seafood”	“Rogede sild”	9.5	9.23	9.50	“Over Average”	“Equal Average”
“Seafood”	“Spegesild”	12	11.11	12.00	“Over Average”	“Equal Average”

## 5.16 Appendix 16: Solution Queries - Question 9

```
--Solution query
WITH cte_kpi AS (
SELECT
    CONCAT(e.first_name, ' ', e.last_name) AS employee_full_name,
    e.title AS employee_title,
    ROUND(
        SUM(d.quantity * d.unit_price)::NUMERIC,
        2) AS total_sale_amount_excluding_discount,
    COUNT(DISTINCT d.order_id) AS total_number_orders,
    COUNT(d.*) AS total_number_entries,
    ROUND(
        SUM(d.discount*(d.quantity * d.unit_price))::NUMERIC,
        2) AS total_discount_amount,
    ROUND(
        SUM((1 - d.discount)*(d.quantity * d.unit_price))::NUMERIC,
        2) AS total_sale_amount_including_discount
FROM orders AS o
INNER JOIN employees AS e
ON o.employee_id = e.employee_id
INNER JOIN order_details AS d
ON d.order_id = o.order_id
INNER JOIN products AS p
```

```

ON d.product_id = p.product_id
GROUP BY
    employee_full_name,
    employee_title
)
SELECT
    employee_full_name,
    employee_title,
    total_sale_amount_excluding_discount,
    total_number_orders,
    total_number_entries,
    ROUND(
        SUM(total_sale_amount_excluding_discount/total_number_entries),
        2) AS average_amount_per_entry,
    ROUND(
        SUM(total_sale_amount_excluding_discount/total_number_orders),
        2) AS average_amount_per_order,
    total_discount_amount,
    total_sale_amount_including_discount,
    SUM(ROUND(
        (total_sale_amount_excluding_discount-total_sale_amount_including_discount)/
        total_sale_amount_excluding_discount*100,
        2)) AS total_discount_percentage

```

```
FROM cte_kpi
GROUP BY
    employee_full_name,
    employee_title,
    total_sale_amount_excluding_discount,
    total_number_orders,
    total_number_entries,
    total_sale_amount_including_discount,
    total_discount_amount
ORDER BY total_sale_amount_including_discount DESC;
```

42

“employee_full_name” title		total_sale_amount_excluding_discount	total_number_orders	total_number_entries	“average_order_size”	“average_order_size”	total_sale_amount_including_discount	total_discount_amount	total_discount_percent
“Margaret Peacock”	“Sales Representative”	250187.45	156	420	595.68	1603.77	17296.60	232890.85	6.91
“Janet Leverling”	“Sales Representative”	213051.30	127	321	663.71	1677.57	10238.46	202812.84	4.81
“Nancy Davolio”	“Sales Representative”	202143.71	123	345	585.92	1643.44	10036.11	192107.60	4.96
“Andrew Fuller”	“Vice President, Sales”	177749.26	96	241	737.55	1851.55	11211.51	166537.76	6.31

“employee_full_name” title		total_sale_amount	total_including_discounts	total_including_beverages	total_including_alcohol	total_including_taxes	total_including_fees	total_including_discounts	total_including_fees
“Laura Callahan”	“Inside Sales Coordinator”	133301.03	104	260	512.70	1281.74	6438.75	126862.28	4.83
“Robert King”	“Sales Representative”	141295.99	72	176	802.82	1962.44	16727.76	124568.23	11.84
“Anne Dodsworth”	“Sales Representative”	82964.00	43	107	775.36	1929.40	5655.93	77308.07	6.82
“Michael Suyama”	“Sales Representative”	78198.10	67	168	465.46	1167.14	4284.97	73913.13	5.48
“Steven Buchanan”	“Sales Manager”	75567.75	42	117	645.88	1799.23	6775.47	68792.28	8.97

## 5.17 Appendix 17: Solution Queries - Question 10

```
--Solution query
WITH cte_kpi AS (
SELECT
    c.category_name,
    CONCAT(e.first_name, ' ', e.last_name) AS employee_full_name,
    ROUND(
        SUM(d.quantity * d.unit_price)::NUMERIC,
        2) AS total_sale_amount_including_discount
FROM employees AS e
INNER JOIN orders AS o
    ON e.employee_id = o.employee_id
INNER JOIN order_details AS d
    ON o.order_id = d.order_id
INNER JOIN products AS p
    ON d.product_id = p.product_id
INNER JOIN categories AS c
    ON c.category_id = p.category_id
GROUP BY
    category_name,
    employee_full_name
)
SELECT
```

```

*,
ROUND(
    SUM(total_sale_amount_including_discount) /
    SUM(SUM(total_sale_amount_including_discount))
    OVER (PARTITION BY employee_full_name) * 100,
    2) AS percentage_of_employee_sales,
ROUND(
    SUM(total_sale_amount_including_discount) /
    SUM(SUM(total_sale_amount_including_discount))
    OVER (PARTITION BY category_name) * 100,
    2) AS percentage_of_category_sales
FROM cte_kpi
GROUP BY
    category_name,
    employee_full_name,
    total_sale_amount_including_discount
ORDER BY
    category_name,
    total_sale_amount_including_discount DESC;

```

---

“category_name”	“employee_full_name”	“total_sale_amount_including_discount”	“percentage_of_employee_sales”	“percentage_of_category_sales”
-----------------	----------------------	--	--------------------------------	--------------------------------

---

“Beverages”	“Margaret Peacock”	52842.35	21.12	18.44
-------------	--------------------	----------	-------	-------

“Beverages”	“Nancy Davolio”	48832.0	24.16	17.04
-------------	-----------------	---------	-------	-------



“category_name”	“employee_full_name”	“total_sale_amount_including_discount”	“percentage_of_employee_sales”	“percentage_of_category_sales”
“Beverages”	“Janet Leverling”	46506.55	21.83	16.23
“Beverages”	“Andrew Fuller”	42029.4	23.65	14.67
“Beverages”	“Robert King”	33517.0	23.72	11.7
“Beverages”	“Anne Dodsworth”	20513.6	24.73	7.16
“Beverages”	“Laura Callahan”	18640.8	13.98	6.51
“Beverages”	“Steven Buchanan”	13517.5	17.89	4.72
“Beverages”	“Michael Suyama”	10127.75	12.95	3.53
“Condiments”	“Margaret Peacock”	25242.45	10.09	22.2
“Condiments”	“Andrew Fuller”	16353.1	9.2	14.38
“Condiments”	“Laura Callahan”	15447.95	11.59	13.59
“Condiments”	“Janet Leverling”	14109.15	6.62	12.41
“Condiments”	“Nancy Davolio”	14033.85	6.94	12.34
“Condiments”	“Anne Dodsworth”	11106.9	13.39	9.77
“Condiments”	“Robert King”	9668.7	6.84	8.5
“Condiments”	“Michael Suyama”	4930.2	6.3	4.34
“Condiments”	“Steven Buchanan”	2802.45	3.71	2.46
“Confections”	“Janet Leverling”	34859.31	16.36	19.68
“Confections”	“Nancy Davolio”	30343.54	15.01	17.13
“Confections”	“Margaret Peacock”	29543.06	11.81	16.68
“Confections”	“Laura Callahan”	22639.5	16.98	12.78
“Confections”	“Andrew Fuller”	21696.95	12.21	12.25
“Confections”	“Robert King”	16754.45	11.86	9.46

“category_name”	“employee_full_name”	“total_sale_amount_including_discount”	“percentage_of_employee_sales”	“percentage_of_category_sales”
“Confections”	“Anne Dodsworth”	8782.94	10.59	4.96
“Confections”	“Michael Suyama”	6990.3	8.94	3.95
“Confections”	“Steven Buchanan”	5489.05	7.26	3.1
“Dairy Products”	“Nancy Davolio”	38012.8	18.8	15.12
“Dairy Products”	“Margaret Peacock”	37047.6	14.81	14.74
“Dairy Products”	“Janet Leverling”	34546.3	16.22	13.75
“Dairy Products”	“Robert King”	30581.5	21.64	12.17
“Dairy Products”	“Andrew Fuller”	25594.7	14.4	10.18
“Dairy Products”	“Steven Buchanan”	23850.4	31.56	9.49
“Dairy Products”	“Anne Dodsworth”	22017.9	26.54	8.76
“Dairy Products”	“Laura Callahan”	21596.2	16.2	8.59
“Dairy Products”	“Michael Suyama”	18083.1	23.12	7.19
“Grains/Cereals”	“Margaret Peacock”	23704.35	9.47	23.53

“category_name”	“employee_full_name”	“total_sale_amount_including_discount”	“percentage_of_employee_sales”	“percentage_of_category_sales”
“Grains/Cereals”	“Janet Leverling”	21898.15	10.28	21.74
“Grains/Cereals”	“Andrew Fuller”	12063.1	6.79	11.98
“Grains/Cereals”	“Laura Callahan”	11723.9	8.8	11.64
“Grains/Cereals”	“Michael Suyama”	9951.1	12.73	9.88
“Grains/Cereals”	“Nancy Davolio”	9173.95	4.54	9.11
“Grains/Cereals”	“Robert King”	6673.5	4.72	6.63
“Grains/Cereals”	“Steven Buchanan”	4233.25	5.6	4.2
“Grains/Cereals”	“Anne Dodsworth”	1305.5	1.57	1.3
“Meat/Poultry”	“Margaret Peacock”	34179.22	13.66	19.18
“Meat/Poultry”	“Andrew Fuller”	32973.51	18.55	18.5
“Meat/Poultry”	“Robert King”	24939.24	17.65	14.0
“Meat/Poultry”	“Janet Leverling”	20921.7	9.82	11.74
“Meat/Poultry”	“Laura Callahan”	17657.5	13.25	9.91
“Meat/Poultry”	“Nancy Davolio”	16346.07	8.09	9.17
“Meat/Poultry”	“Steven Buchanan”	11869.4	15.71	6.66
“Meat/Poultry”	“Anne Dodsworth”	9920.36	11.96	5.57
“Meat/Poultry”	“Michael Suyama”	9381.8	12.0	5.27
“Produce”	“Nancy Davolio”	20454.45	10.12	19.43
“Produce”	“Margaret Peacock”	18586.75	7.43	17.66
“Produce”	“Janet Leverling”	12718.15	5.97	12.08
“Produce”	“Michael Suyama”	12323.9	15.76	11.71
“Produce”	“Laura Callahan”	12146.8	9.11	11.54

“category_name”	“employee_full_name”	“total_sale_amount_including_discount”	“percentage_of_employee_sales”	“percentage_of_category_sales”
“Produce”	“Robert King”	11552.4	8.18	10.97
“Produce”	“Andrew Fuller”	9476.8	5.33	9.0
“Produce”	“Steven Buchanan”	7605.1	10.06	7.22
“Produce”	“Anne Dodsworth”	404.25	0.49	0.38
“Seafood”	“Margaret Peacock”	29041.67	11.61	20.51
“Seafood”	“Janet Leverling”	27491.99	12.9	19.41
“Seafood”	“Nancy Davolio”	24947.05	12.34	17.62
“Seafood”	“Andrew Fuller”	17561.7	9.88	12.4
“Seafood”	“Laura Callahan”	13448.38	10.09	9.5
“Seafood”	“Anne Dodsworth”	8912.55	10.74	6.29
“Seafood”	“Robert King”	7609.2	5.39	5.37
“Seafood”	“Michael Suyama”	6409.95	8.2	4.53
“Seafood”	“Steven Buchanan”	6200.6	8.21	4.38