

COM2058 Project 1

Name, Surname: Elvin Huseynli

Date: 21/03/2022

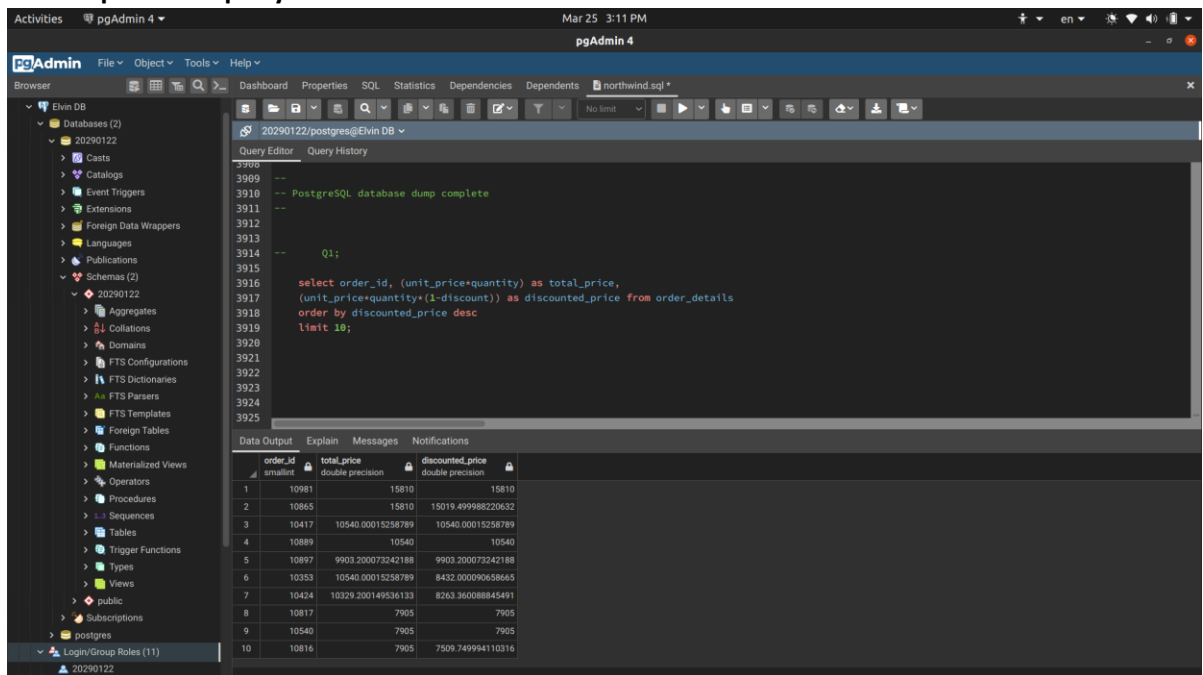
Student Number: 20290122

Sign: 

Task B:

The main objective of this task is to write queries and run them in pgAdmin4 using PostgreSQL. We are expected to add screenshots for each subtask, including queries and their outputs in table format.

(1) The output of 1st query:



The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, including the '20290122' database. The main window shows a SQL query in the Query Editor, which has been executed. The output is displayed in a table format below the query editor.

```
-- PostgreSQL database dump complete
--
--      Q1;
--
select order_id, (unit_price*quantity) as total_price,
       (unit_price*quantity*(1-discount)) as discounted_price from order_details
order by discounted_price desc
limit 10;
```

order_id	total_price	discounted_price
1	10981	15810
2	10865	15810
3	10417	10540.00015258789
4	10889	10540
5	10897	9903.200073242188
6	10353	10540.00015258789
7	10424	10329.200149536133
8	10817	7905
9	10540	7905
10	10816	7509.749994110316

(2) The output of 2nd query:

pgAdmin 4

Query Editor

```
3916 create view ret as
3917 (select shipped_date, od.order_id, (od.unit_price*od.quantity) as total_price,
3918 extract(year from shipped_date) as years from orders
3919 inner join order_details od on od.order_id = orders.order_id
3920 where shipped_date between '1997-12-30' and '1998-01-05' order by shipped_date asc, order_id asc)
3921 union all
3922 (select shipped_date, od.order_id, sum(od.unit_price*od.quantity) total_price,
3923 extract(year from shipped_date) as years from orders
3924 inner join order_details od on od.order_id = orders.order_id
3925 group by shipped_date, od.order_id, years order by order_id asc);
3926 (select shipped_date, od.order_id, sum(od.unit_price*od.quantity) total_price,
3927 extract(year from shipped_date)::numeric as years from orders
3928 inner join order_details od on od.order_id = orders.order_id
3929 where shipped_date between '1997-12-30' and '1998-01-05'
3930 group by shipped_date, od.order_id, years order by shipped_date asc, order_id asc)
3931 union all
3932 (select ret.shipped_date, ret.order_id, ret.total_price, ret.years::numeric from ret
3933 where ret.shipped_date is null and ret.total_price=4000
3934 group by ret.shipped_date, ret.order_id, ret.years, ret.total_price)
3935 ----
```

Data Output

shipped_date	order_id	total_price	years
1997-12-31	10789	3687.00045776367	1997
1997-12-31	10792	399.8499994277954	1997
1997-12-31	10801	4035.800018310547	1997
1998-01-01	10791	1926.0600051876883	1998
1998-01-02	10771	344	1998
1998-01-02	10794	393.4499988550308	1998
1998-01-02	10802	3923.7499809265137	1998
1998-01-05	10797	420	1998
1998-01-05	10798	446.5999908472956	1998
1998-01-05	10799	1285	1998
1998-01-05	10800	1632.14999864886	1998
1998-01-05	10806	572.0999984741211	1998
null	11008	4903.498993188477	null
null	11072	5217.999984741211	null

(3) The output of 3rd query:

pgAdmin 4

Query Editor

```
3913
3914
3915
3916
3917
3918 -- Q3:
3919 select categories.category_name, products.category_id, products.product_name, products.product_id,
3920 products.unit_price, products.units_in_stock, products.units_on_order,
3921 products.reorder_level, products.discontinued
3922 from categories
3923 right join products on products.category_id = categories.category_id
3924 where discontinued=0 and reorder_level>20 and units_on_order=0
3925 order by product_name;
```

Data Output

category_name	category_id	product_name	product_id	unit_price	units_in_stock	units_on_order	reorder_level	discontinued
Seafood	8	Boston Crab Meat	40	18.4	123	0	30	0
Grains/Cereals	5	Filo Mix	52	7	38	0	25	0
Condiments	2	Grandma's Boysenberry Spread	6	25	120	0	25	0
Grains/Cereals	5	Gustaf's Kaviar	22	21	104	0	25	0
Confections	3	NuNuCa Nuß-Nougat-Creme	25	14	76	0	30	0
Beverages	1	Rönkäbräu Kasteleibier	75	7.75	125	0	25	0
Confections	3	Schoggi Schokolade	27	43.9	49	0	30	0
Condiments	2	Simp. d'Épice	41	28.5	113	0	25	0
Grains/Cereals	5	Tuusula	22	9	61	0	25	0
Confections	3	Valkoinen suklaa	50	16.25	65	0	30	0

(4) The output of 4th query:

Activities pgAdmin 4 Mar 25 1:34 PM

pgAdmin 4

Dashboard Properties SQL Statistics Dependencies Dependents northwind.sql

Query Editor Query History

```

3917 (select orders.ship_name, orders.ship_country, customers.customer_id, customers.company_name,
3918 concat(employees.first_name, ' ', employees.last_name) as salesperson,
3919 shippers.phone, products.product_id, products.product_name, orders.freight
3920   right join employees on employees.employee_id = orders.employee_id
3921   right join customers on customers.customer_id = orders.customer_id
3922   right join shippers on shippers.shipper_id = orders.ship_via
3923   left join order_details on orders.order_id = order_details.order_id
3924   left join products on products.product_id = order_details.product_id
3925 where orders.ship_country like 'I%' and ship_name like 'MS' and freight > 70 order by salesperson asc)
3926 union all
3927 (select orders.ship_name, orders.ship_country, customers.customer_id, customers.company_name,
3928 concat(employees.first_name, ' ', employees.last_name) as salesperson,
3929 shippers.phone, products.product_id, products.product_name, orders.freight
3930   right join employees on employees.employee_id = orders.employee_id
3931   right join customers on customers.customer_id = orders.customer_id
3932   right join shippers on shippers.shipper_id = orders.ship_via
3933   left join order_details on orders.order_id = order_details.order_id
3934   left join products on products.product_id = order_details.product_id
3935 where orders.ship_country like 'G%' and ship_name like 'MS' and freight > 70 order by salesperson asc);
3936

```

Data Output Explain Messages Notifications

ship_name	ship_country	customer_id	company_name	salesperson	phone	product_id	product_name	freight
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Andrew Fuller	(503) 555-3199	2	Chang	76.33
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Andrew Fuller	(503) 555-3199	62	Laughing Lumberjack Lager	76.33
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Andrew Fuller	(503) 555-9831	26	Gumbler Gumbärschen	155.97
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Andrew Fuller	(503) 555-9831	42	Singaporean Hokkaid Fried Mee	155.97
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Andrew Fuller	(503) 555-9831	49	Mazatlan	155.97
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Margaret Peacock	(503) 555-9931	36	Wheat Stiff	70.09
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Margaret Peacock	(503) 555-9931	39	Charreusse verte	70.09
Magazzini Alimentari Riuniti	Italy	MAGAA	Magazzini Alimentari Riuniti	Margaret Peacock	(503) 555-9931	72	Mozzarella di Giovanni	70.09
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Andrew Fuller	(503) 555-9931	28	Rossini Seawinkles	125.77
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Andrew Fuller	(503) 555-9931	62	Tarte au sucre	125.77
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Steven Buchanan	(503) 555-9831	59	Raclette Courferault	127.34
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Steven Buchanan	(503) 555-9831	63	Vergespread	127.34
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Steven Buchanan	(503) 555-9831	72	Mozzarella di Giovanni	127.34
Morgenstern Gesundkost	Germany	MOROK	Morgenstern Gesundkost	Steven Buchanan	(503) 555-9831	76	Lakkakhoon	127.34

(5) The output of 5th query:

Activities pgAdmin 4 Mar 25 1:37 PM

pgAdmin 4

Dashboard Properties SQL Statistics Dependencies Dependents northwind.sql

Query Editor Query History

```

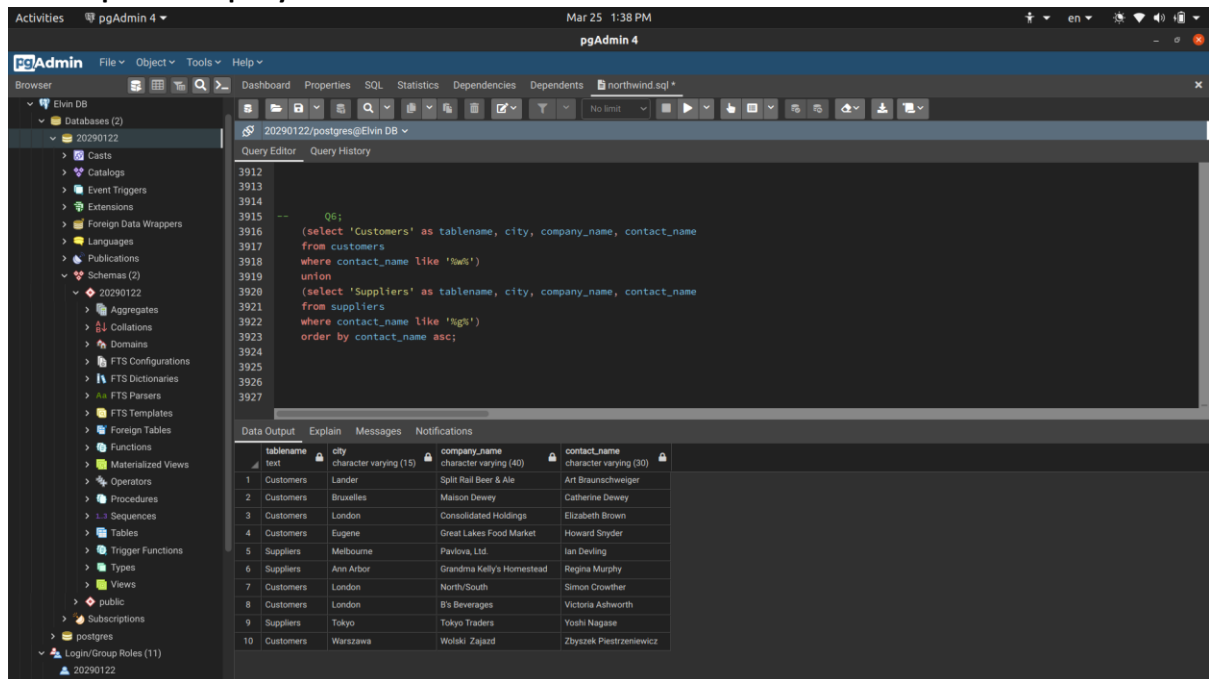
3915
3916 alter table orders
3917 add "Year 1999" double precision, add "Year 1998" double precision, add "Year 1997" double precision, add "Year 1996" double precision;
3918 update orders set "Year 1999" = 0; update orders set "Year 1998" = 0; update orders set "Year 1997" = 0; update orders set "Year 1996" = 0;
3919 (select od.product_id, orders.customer_id, extract(year from orders.order_date) as "Year",
3920   "Year 1999", "Year 1998", "Year 1997", (od.unit_price*od.quantity*(1-od.discount)) as "Year 1996" from orders
3921   right join order_details od on od.order_id = orders.order_id
3922   where od.product_id < 5 and orders.customer_id like 'E%' and extract(year from orders.order_date) = 1996 order by "Year", od.product_id)
3923 union all
3924 (select od.product_id, orders.customer_id, extract(year from orders.order_date) as "Year",
3925   "Year 1999", "Year 1998", (od.unit_price*od.quantity*(1-od.discount)) as "Year 1997", "Year 1996" from orders
3926   right join order_details od on od.order_id = orders.order_id
3927   where od.product_id < 5 and orders.customer_id like 'E%' and extract(year from orders.order_date) = 1997 order by "Year", od.product_id)
3928 union all
3929 (select od.product_id, orders.customer_id, extract(year from orders.order_date) as "Year",
3930   "Year 1999", (od.unit_price*od.quantity*(1-od.discount)) as "Year 1998", "Year 1997", "Year 1996" from orders
3931   right join order_details od on od.order_id = orders.order_id
3932   where od.product_id < 5 and orders.customer_id like 'E%' and extract(year from orders.order_date) = 1998 order by "Year", od.product_id)
3933 union all
3934 (select od.product_id, orders.customer_id, extract(year from orders.order_date) as "Year",
3935   (od.unit_price*od.quantity*(1-od.discount)) as "Year 1999", "Year 1998", "Year 1997", "Year 1996" from orders
3936   right join order_details od on od.order_id = orders.order_id
3937   where od.product_id < 5 and orders.customer_id like 'E%' and extract(year from orders.order_date) = 1999 order by "Year", od.product_id)
3938
3939
3940

```

Data Output Explain Messages Notifications

product_id	customer_id	Year	Year 1999	Year 1998	Year 1997	Year 1996
1	2	1996	0	0	0	607.999999105629
2	3	1997	0	0	176.999999701676	0
3	4	1997	0	0	880	0
4	1	1998	0	327.8	0	0
5	2	1998	0	152	0	0
6	3	1998	0	250	0	0

(6) The output of 6th query:



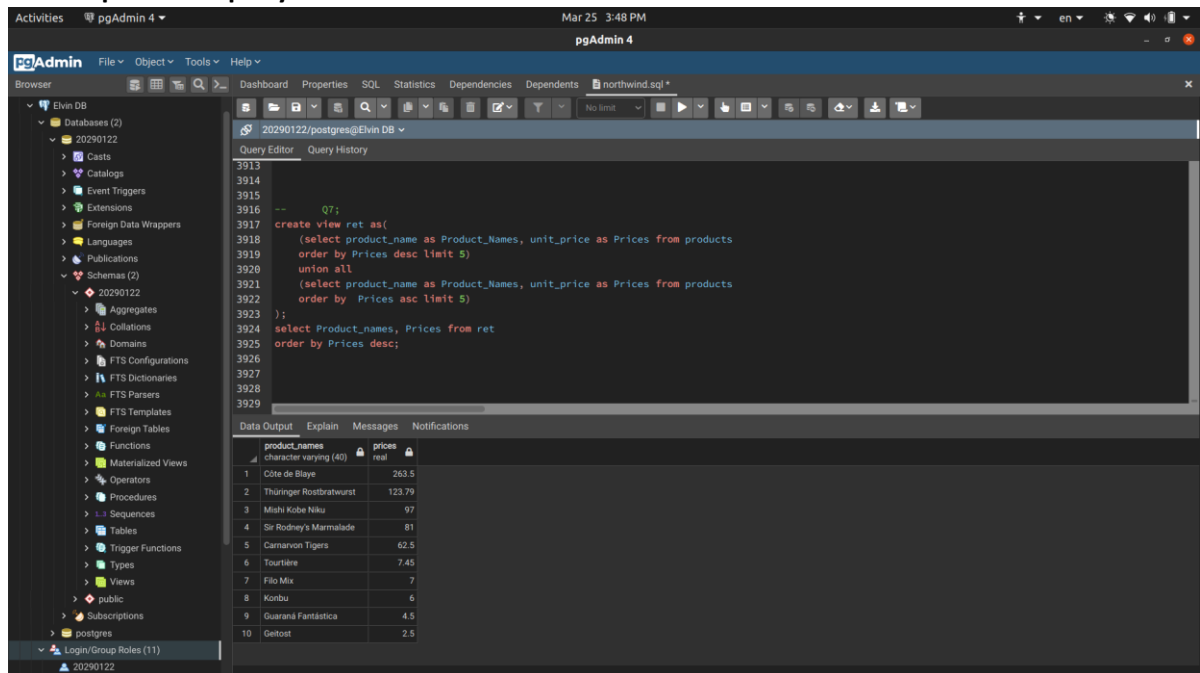
The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure for 'Elvin DB', including 'Databases (2)', 'Schemas (2)', and '20290122'. The main pane shows the 'Query Editor' for '20290122/postgres@Elvin DB'. The query is as follows:

```
-- Q6;
(select 'Customers' as tablename, city, company_name, contact_name
from customers
where contact_name like '%a%')
union
(select 'Suppliers' as tablename, city, company_name, contact_name
from suppliers
where contact_name like '%g%')
order by contact_name asc;
```

The 'Data Output' tab shows the results of the query:

tablename	city	company_name	contact_name
Customers	Lander	Split Rail Beer & Ale	Art Braunschweiger
Customers	Brussels	Maison Dewey	Catherine Dewey
Customers	London	Consolidated Holdings	Elizabeth Brown
Customers	Eugene	Great Lakes Food Market	Howard Snyder
Suppliers	Melbourne	Pavlova, Ltd.	Ian Devling
Suppliers	Ann Arbor	Grandma Kelly's Homestead	Regina Murphy
Customers	London	North/South	Simon Crowther
Customers	London	B's Beverages	Victoria Ashworth
Suppliers	Tokyo	Tokyo Traders	Yoshi Nagase
Customers	Warszawa	Wolaki Zajad	Zbyszek Piestrzeniewicz

(7) The output of 7th query:



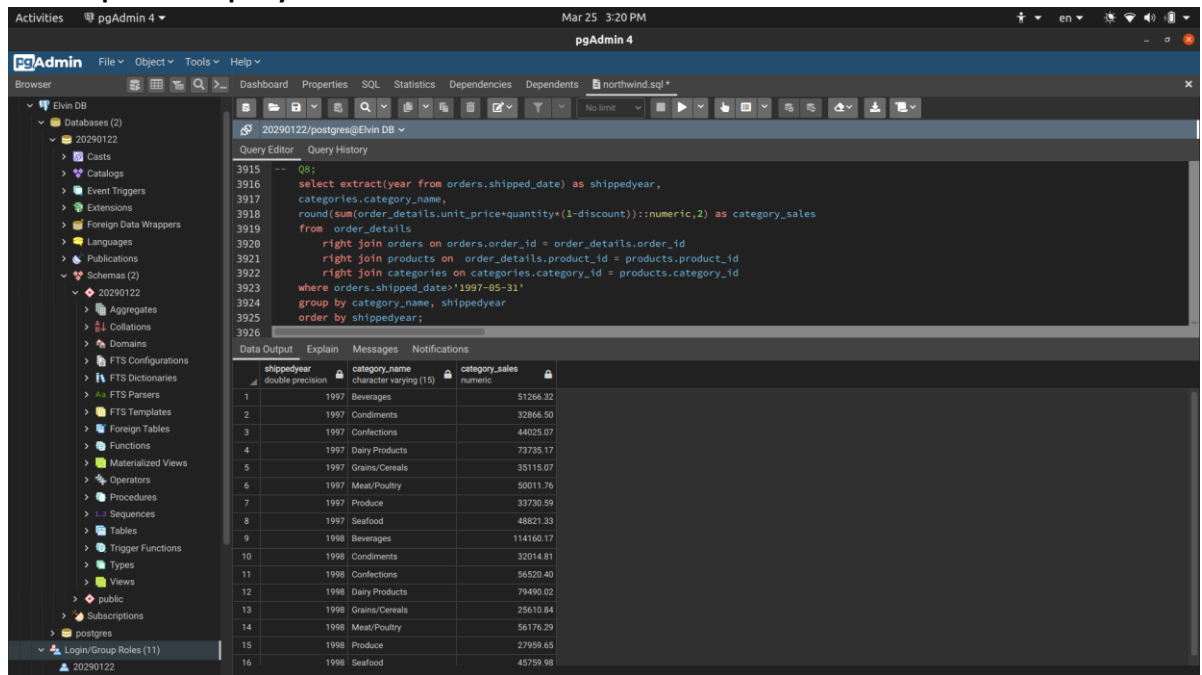
The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure for 'Elvin DB', including 'Databases (2)', 'Schemas (2)', and '20290122'. The main pane shows the 'Query Editor' for '20290122/postgres@Elvin DB'. The query is as follows:

```
-- Q7;
create view ret as(
(select product_name as Product_Names, unit_price as Prices from products
order by Prices desc limit 5)
union all
(select product_name as Product_Names, unit_price as Prices from products
order by Prices asc limit 5)
);
select Product_names, Prices from ret
order by Prices desc;
```

The 'Data Output' tab shows the results of the query:

product_names	prices
Côte de Blaye	263.5
Thüringer Rostbratwurst	123.79
Mishi Kobe Niku	97
Sir Rodney's Marmalade	81
Camaron de Tigers	62.5
Touffiere	7.45
Filo Mix	7
Konbu	6
Guaraná Fantástica	4.5
Getrost	2.5

(8) The output of 8th query:



The screenshot shows the pgAdmin 4 interface. The left sidebar displays a tree view of the database structure, including 'Elvin DB', 'Databases (2)', '20290122', 'Catalogs', 'Event Triggers', 'Extensions', 'Foreign Data Wrappers', 'Languages', 'Publications', 'Schemas (2)', '20290122', 'Aggregates', 'Collations', 'Domains', 'FTS Configurations', 'FTS Dictionaries', 'FTS Parsers', 'FTS Templates', 'Foreign Tables', 'Functions', 'Materialized Views', 'Operators', 'Procedures', 'Sequences', 'Tables', 'Trigger Functions', 'Types', 'Views', 'public', 'Subscriptions', 'postgres', and 'Login/Group Roles (11)'. The main pane shows the 'Query Editor' for the '20290122/postgres@Elvin DB' connection. The query is as follows:

```
-- Q8;
select extract(year from orders.shipped_date) as shippedyear,
categories.category_name,
round(sum(order_details.unit_price*quantity*(1-discount))::numeric,2) as category_sales
from order_details
right join orders on orders.order_id = order_details.order_id
right join products on order_details.product_id = products.product_id
right join categories on categories.category_id = products.category_id
where orders.shipped_date > '1997-05-31'
group by category_name, shippedyear
order by shippedyear;
```

The 'Data Output' tab shows the results of the query, which are 16 rows of data. The columns are 'shippedyear' (double precision), 'category_name' (character varying (15)), and 'category_sales' (numeric). The data is as follows:

shippedyear	category_name	category_sales
1997	Beverages	51266.32
1997	Condiments	32866.50
1997	Confections	44025.07
1997	Dairy Products	73735.17
1997	Grains/Cereals	35115.07
1997	Meat/Poultry	50011.76
1997	Produce	33730.59
1997	Seafood	48821.33
1998	Beverages	114160.17
1998	Condiments	32014.81
1998	Confections	56520.40
1998	Dairy Products	79490.02
1998	Grains/Cereals	25616.84
1998	Meat/Poultry	56176.29
1998	Produce	27959.65
1998	Seafood	45759.98

Task C:

The main objective of this task is to apply relational algebra to the queries we have written in Task B section. We are expected to use them as efficiently as possible.

① $OD \leftarrow \sigma_{\text{order-details}}$

$RESULT \leftarrow \rho_{(\text{order-id}, \text{total-price}, \text{discounted-price})} (\pi_{\text{order-id}, (\text{unit-price} * \text{quantity}), (\text{unit-price} * \text{quantity} * (1 - \text{discount}))} (OD))$

② $FIRST \leftarrow \sigma_{'1997-12-30' < \text{shipped-date} < '1998-01-05'} (\text{orders})$

$FIRST_RES \leftarrow \rho_{(\text{shipped-date}, \text{order-id}, \text{total-price}, \text{years})} (\pi_{\text{shipped-date}, \text{od.order-id}, (\text{od.unit-price} * \text{od.quantity})} (FIRST \bowtie_{\text{od.order-id} = \text{orders.order-id}} (\text{order-details od})))$

$SEC \leftarrow \sigma_{\text{shipped-date} = \text{NULL} \text{ AND } \text{total-price} > 4000} (\text{orders})$

$SEC_RES \leftarrow \rho_{(\text{shipped-date}, \text{order-id}, \text{total-price}, \text{years})} (\pi_{\text{shipped-date}, \text{od.order-id}, (\text{od.unit-price} * \text{od.quantity})} (SEC \bowtie_{\text{od.order-id} = \text{orders.order-id}} (\text{order-details od})))$

$RESULT \leftarrow (FIRST_RES \cup SEC_RES)$

③ $CAT \leftarrow \sigma_{\text{discontinued}=0 \text{ AND } \text{reorder-level} > 20 \text{ AND } \text{units-on-order} = 0} (\text{categories})$

$RESULT \leftarrow (\pi_{\text{categories.category-name}, \text{products.category-id}, \text{products.product-name}, \text{products.product-id}, \text{products.unit-price}, \text{products.units-in-stock}, \text{products.units-on-order}, \text{products.reorder-level}, \text{products.discontinued}} (CAT \bowtie_{\text{products.category-id} = \text{categories.category-id}} (\text{products})))$

⑦ $FR \leftarrow \pi_{\text{product-name}, \text{unit-price}} (\text{products}) \quad \{ \text{for top 5} \}$

$SR \leftarrow \pi_{\text{product-name}, \text{unit-price}} (\text{products}) \quad \{ \text{for top least 5} \}$

⑧ $FA \leftarrow \sigma_{\text{orders.shipped-date} > '1997-05-31'} (\text{order-details})$

$RESULT \leftarrow \rho_{\text{shipped-year}, \text{category-name}, \text{category-sales}} (\pi_{\text{shipped-date}, \text{categories.category-name}, \text{unit-price} * \text{quantity} * (1 - \text{discount})}$

$(FA \bowtie_{\text{orders.order-id} = \text{order-details.order-id}} (\text{orders})) (\text{order-details} \bowtie_{\text{order-details.product-id} = \text{products.product-id}} (\text{products}))$
 $(\text{products} \bowtie_{\text{products.category-id} = \text{categories.category-id}} (\text{categories}))$

Task D:

The main objective of this task is to connect to the database using Python and its library. We are expected to write the queries in Python and add the screenshots of the outputs of them.

(1) The output of 1st query:

```
Activities Terminal Mar 25 3:36 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-1

OrderId  TotalPrice  TotalwithDiscountPrice
10981    15810.0      15810.0
10865    15810.0      15819.499988220632
10417    10540.00015258789  10540.00015258789
10889    10540.0      10540.0
10897    9903.200073242188  9903.200073242188
10353    10540.00015258789  8432.000090658665
10424    10329.200149536133  8263.360088845491
10817    7985.0       7985.0
10540    7985.0       7985.0
10816    7985.0       7569.749994110316

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```

(2) The output of 2nd query:

```
Activities Terminal Mar 25 3:37 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-2

ShippedDate  OrderId  TotalPrice  Years
1997-12-31   10789    3687.000045776367  1997
1997-12-31   10792    399.8499994277954  1997
1997-12-31   10801    4035.800018310547  1997
1998-01-01   10791    1926.0600051879883  1998
1998-01-02   10771    344.0          1998
1998-01-02   10794    393.4499988555908  1998
1998-01-02   10802    3923.7499809265137  1998
1998-01-05   10797    420.0          1998
1998-01-05   10798    446.59999084472656  1998
1998-01-05   10799    1585.0         1998
1998-01-05   10800    1632.149998664856  1998
1998-01-05   10806    572.0999984741211  1998
None         11008    4903.499893188477  None
None         11072    5217.999984741211  None

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```


(3) The output of 3rd query:

```
Activities Terminal Mar 25 3:41 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-3

CategoryName  CategoryId  ProductName  ProductId  UnitPrice  UnitsInStock  UnitsOnOrder  ReorderLevel  Discontinued
Seafood       8           Boston Crab Meat  40         18.4       123           0             30            0
Grains/Cereals 5           Fllo Mix  52         7.0        38            0             25            0
Condiments    2           Grandma's Boysenberry Spread  6          25.0       120           0             25            0
Grains/Cereals 5           Gustaf's Knäckebröd  22         21.0       104           0             25            0
Confections   3           NuNuCa Nuü-Nougat-Creme  25         14.0       76            0             30            0
Beverages     1           Rhönbräu Klosterbier  75         7.75       125           0             25            0
Confections   3           Schoggi Schokolade  27         43.9       49            0             30            0
Condiments    2           Sirop d'érable  61         28.5       113           0             25            0
Grains/Cereals 5           Tunnbröd  23         9.0        61            0             25            0
Confections   3           Valkoinen suklaa  50         16.25      65            0             30            0

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```

(4) The output of 4th query:

```
Activities Terminal Mar 25 3:41 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-4

ShipName      ShipCountry  CustomerId  CompanyName  SalesPerson  Phone  ProductId  ProductName  Freight
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Andrew Fuller (503) 555-3199 2 Chang 76.33
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Andrew Fuller (503) 555-3199 67 Laughing Lumberjack Lager 76.33
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Andrew Fuller (503) 555-9831 26 Gunbar Gumbibärchen 155.97
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Andrew Fuller (503) 555-9831 42 Singaporean Hokkien Fried Mee 155.97
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Andrew Fuller (503) 555-9831 49 Makilaku 155.97
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Margaret Peacock (503) 555-9931 36 Inlaga Sill 70.09
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Margaret Peacock (503) 555-9931 39 Chartreuse verte 70.09
Magazzini Alimentari Riuniti Italy        MAGAA      Magazzini Alimentari Riuniti Margaret Peacock (503) 555-9931 72 Mozzarella di Giovanni 70.09
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Andrew Fuller (503) 555-9931 28 Rösle Sauerkraut 125.77
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Andrew Fuller (503) 555-9931 62 Tarte au sucre 125.77
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Steven Buchanan (503) 555-9831 59 Raclette Courdavault 127.34
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Steven Buchanan (503) 555-9831 63 Veggie-spread 127.34
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Steven Buchanan (503) 555-9831 72 Mozzarella di Giovanni 127.34
Morgenstern Gesundkost Germany     NORCK      Morgenstern Gesundkost Steven Buchanan (503) 555-9831 76 Lakkalikööri 127.34

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```


(5) The output of 5th query:

```
Activities Terminal Mar 25 3:42 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-5

ProductId CustomerId Year Year 1999 Year 1998 Year 1997 Year 1996
2 ERNSH 1996 0.0 0.0 0.0 607.9999990185629
3 ERNSH 1997 0.0 0.0 179.99999970197678 0.0
4 EASTC 1997 0.0 0.0 550.0 0.0
1 EASTC 1998 0.0 337.5 0.0 0.0
2 ERNSH 1998 0.0 152.0 0.0 0.0
3 ERNSH 1998 0.0 250.0 0.0 0.0

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```

(6) The output of 6th query:

```
Activities Terminal Mar 25 3:42 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-6

TableName City CompanyName ContactName
Customers Lander Split Rail Beer & Ale Art Braunschweiler
Customers Bruxelles Maison Dewey Catherine Dewey
Customers London Consolidated Holdings Elizabeth Brown
Customers Eugene Great Lakes Food Market Howard Snyder
Suppliers Melbourne Pavlova, Ltd. Ian Devling
Suppliers Ann Arbor Grandma Kelly's Homestead Regina Murphy
Customers London North/South Simon Crowther
Customers London B's Beverages Victoria Ashworth
Suppliers Tokyo Tokyo Traders Yoshi Nagase
Customers Warszawa Wolski Zajazd Zbyszek Plestrzeniewicz

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```

(7) The output of 7th query:

```
Activities Terminal Mar 25 3:50 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-7

ProductName      Prices
Côte de Blaye    263.5
Thüringer Rostbratwurst 123.79
Mishl Kobe Niku  97.0
Sir Rodney's Marmalade 81.0
Carnarvon Tigers 62.5
Tourtière        7.45
Pilo Mix         7.0
Konbu            6.0
Guaraná Fantástica 4.5
Geltost          2.5

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```

(8) The output of 8th query:

```
Activities Terminal Mar 25 3:51 PM
elvin@elvin: ~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D
elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$ python3 20290122_Project_1.py
Question-8

ShippedYear      CategoryName      CategorySales
1997.0            Beverages         51266.32
1997.0            Condiments        32866.58
1997.0            Confections       44025.07
1997.0            Dairy Products    73735.17
1997.0            Grains/Cereals    35115.07
1997.0            Meat/Poultry      50811.76
1997.0            Produce           33730.59
1997.0            Seafood           48821.33
1998.0            Beverages         114169.17
1998.0            Condiments        32014.81
1998.0            Confections       56520.40
1998.0            Dairy Products    79490.02
1998.0            Grains/Cereals    25610.04
1998.0            Meat/Poultry      56176.29
1998.0            Produce           27959.65
1998.0            Seafood           45759.98

elvin@elvin:~/Desktop/4th Term/COM2058-Database Management/Projects/Solutions/Project1/Task D$
```