SQL-Capstone Project

Analysis of Amazon sales to understand the different factors that affect sales of the branches.

Nikhil Karaka

Agenda

1 Introduction

2 Data Wrangling And Feature Engineering

3 Business questions

4 Analysis

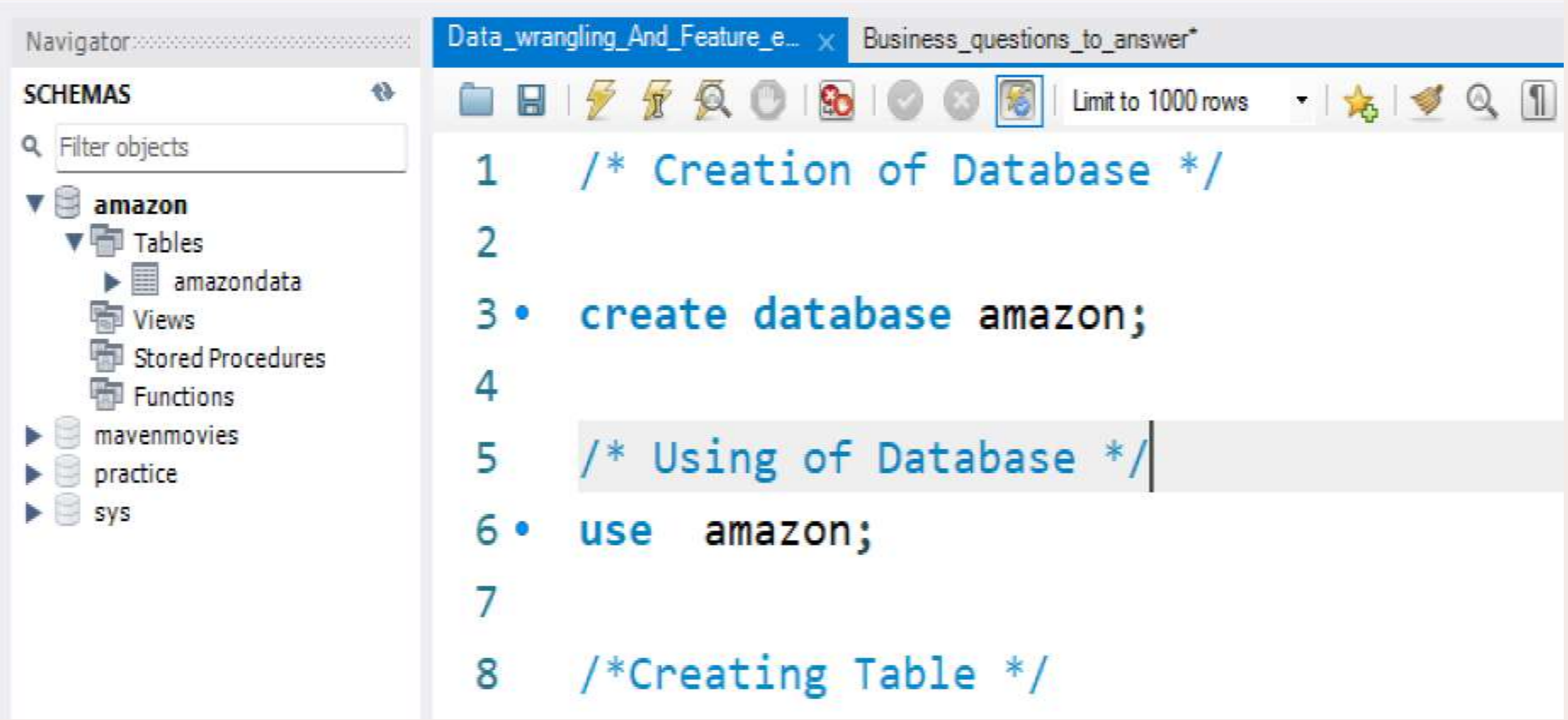
5 Insights and Summary

Introduction

This dataset contains sales transactions from three different branches of Amazon, respectively located in Mandalay, Yangon and Naypyitaw. The data contains 20 columns and 1000 rows.

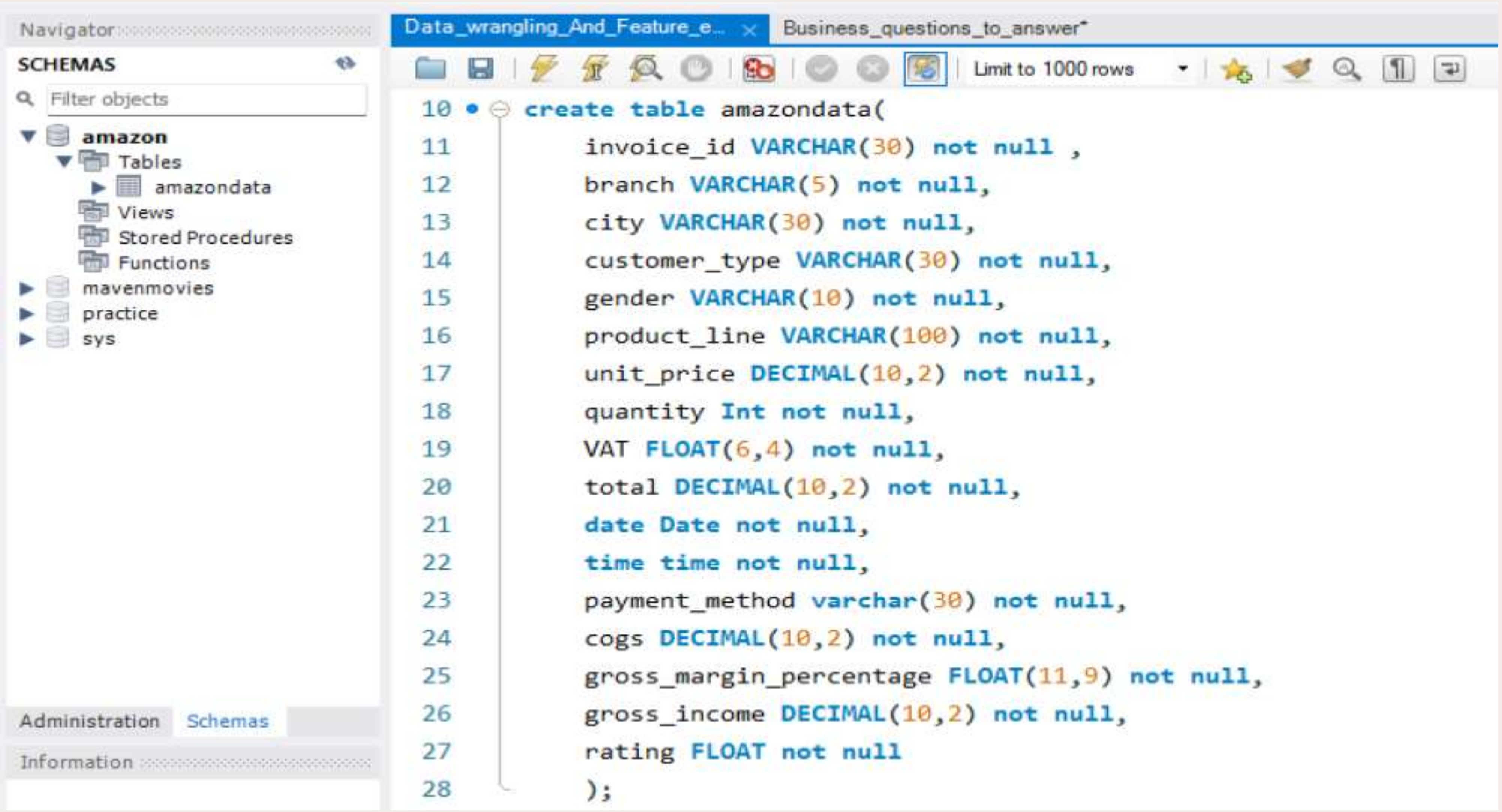
Data Wrangling

- Building a database

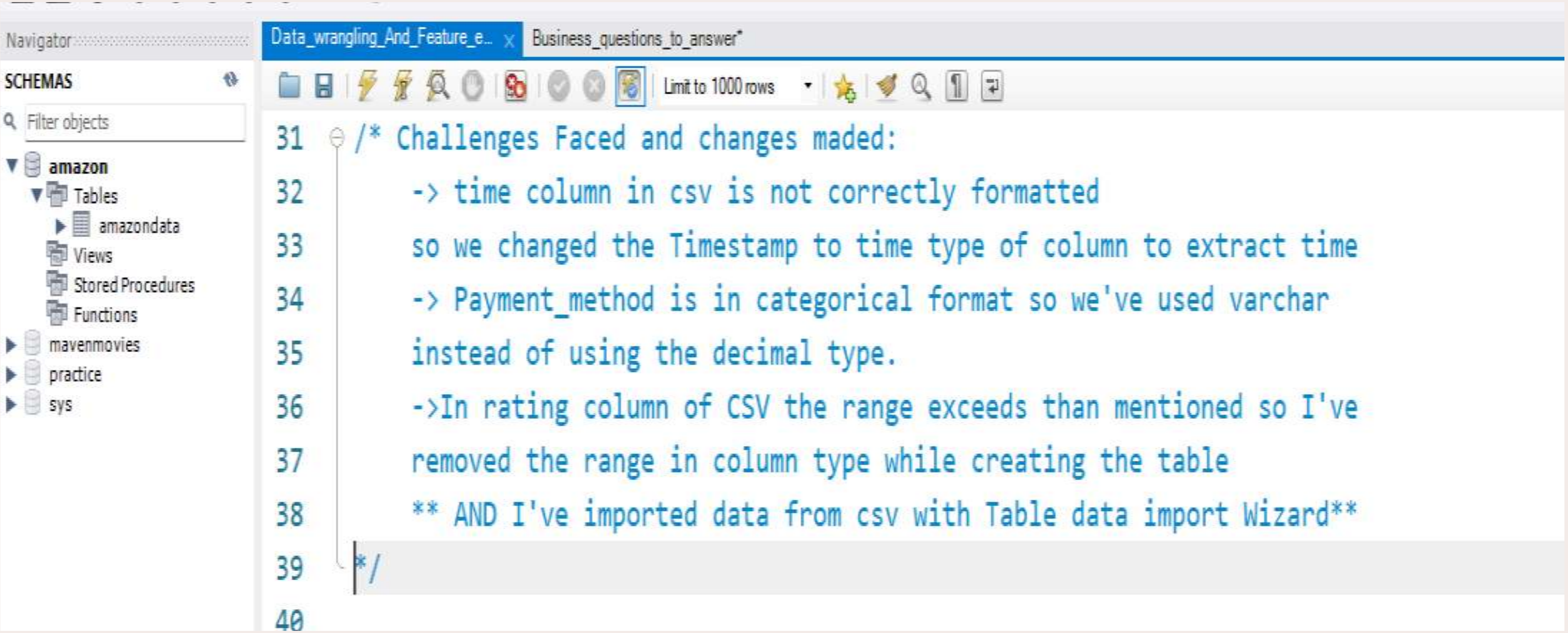


```sql
/* Creation of Database */

create database amazon;

/* Using of Database */
use  amazon;

/*Creating Table */
```

- Creating a table and Insert the data



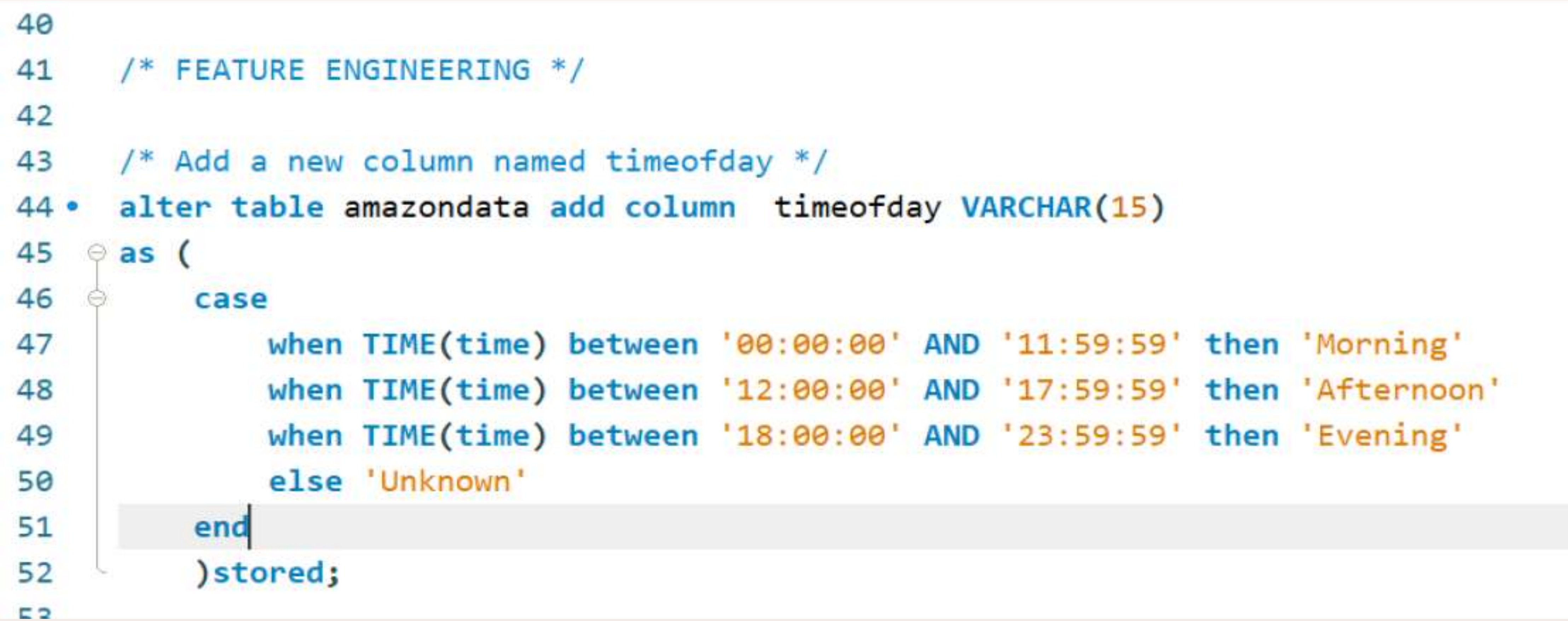
```
create table amazondata(
        invoice_id VARCHAR(30) not null ,
        branch VARCHAR(5) not null,
        city VARCHAR(30) not null,
        customer_type VARCHAR(30) not null,
        gender VARCHAR(10) not null,
        product_line VARCHAR(100) not null,
        unit_price DECIMAL(10,2) not null,
        quantity Int not null,
        VAT FLOAT(6,4) not null,
        total DECIMAL(10,2) not null,
        date Date not null,
        time time not null,
        payment_method varchar(30) not null,
        cogs DECIMAL(10,2) not null,
        gross_margin_percentage FLOAT(11,9) not null,
        gross_income DECIMAL(10,2) not null,
        rating FLOAT not null
        );
```



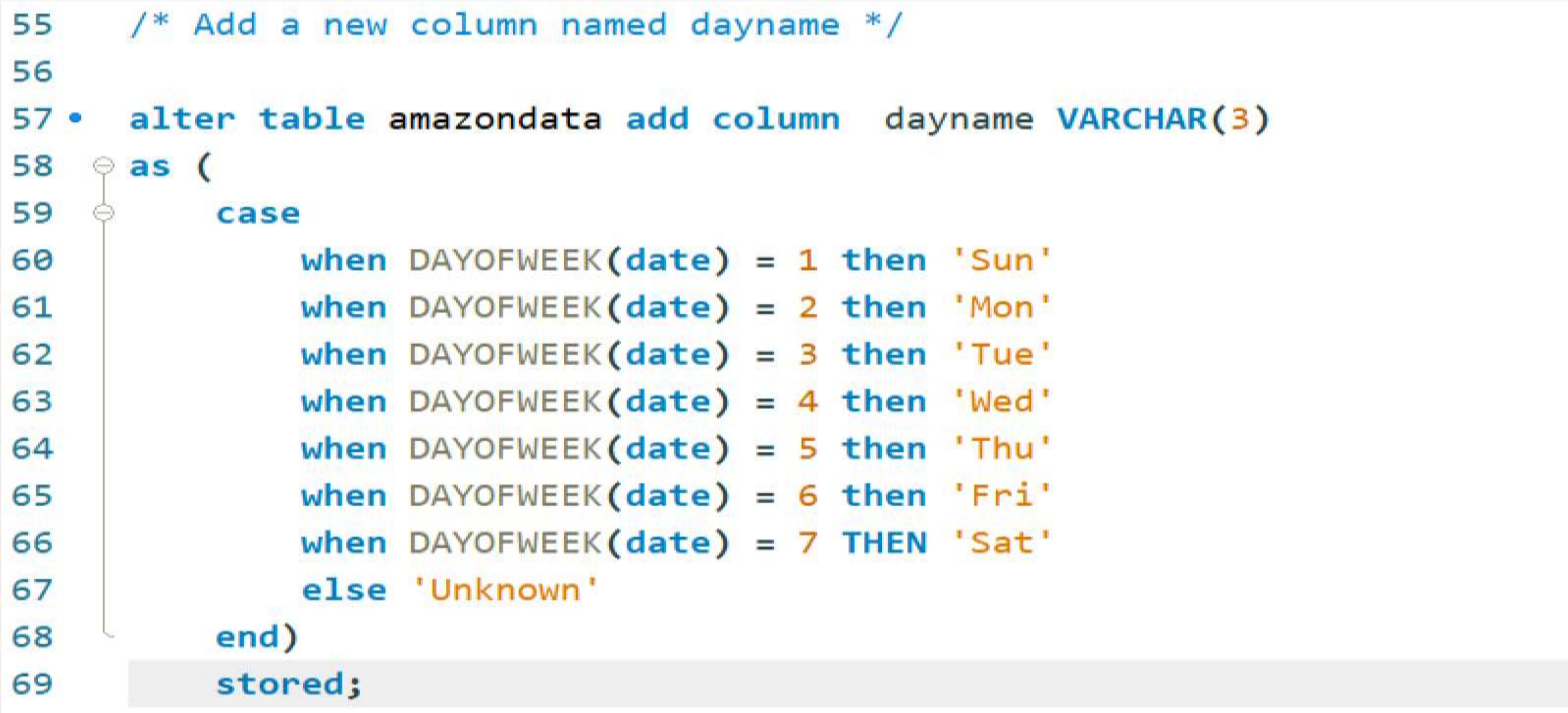
```
/* Challenges Faced and changes maded:
        -> time column in csv is not correctly formatted
        so we changed the Timestamp to time type of column to extract time
        -> Payment_method is in categorical format so we've used varchar
        instead of using the decimal type.
        ->In rating column of CSV the range exceeds than mentioned so I've
        removed the range in column type while creating the table
        ** AND I've imported data from csv with Table data import Wizard**
*/
```

- There are no null values in our database as in creating the tables, we set NOT NULL for each field, hence null values are filtered out.

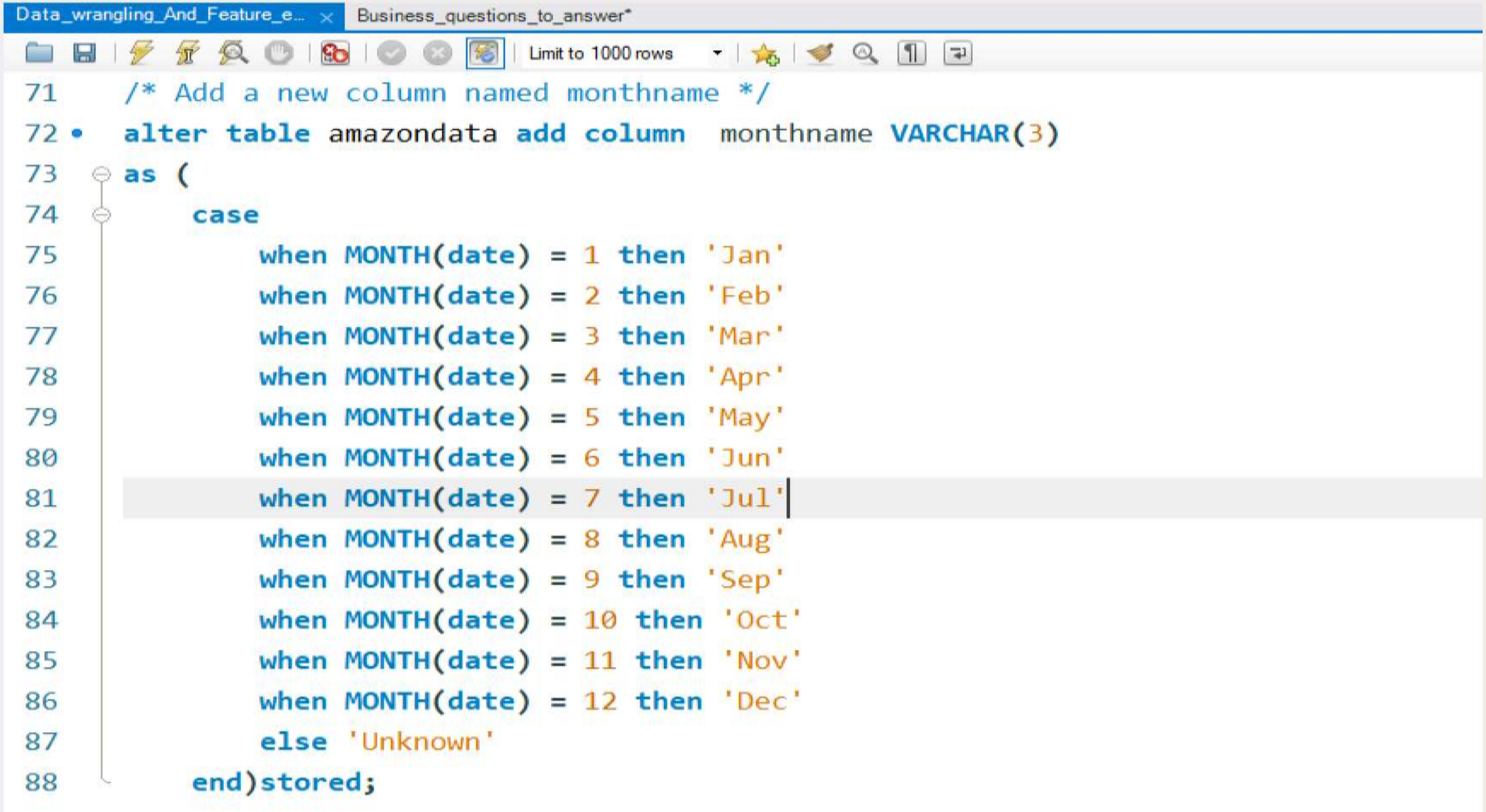
Feature Engineering



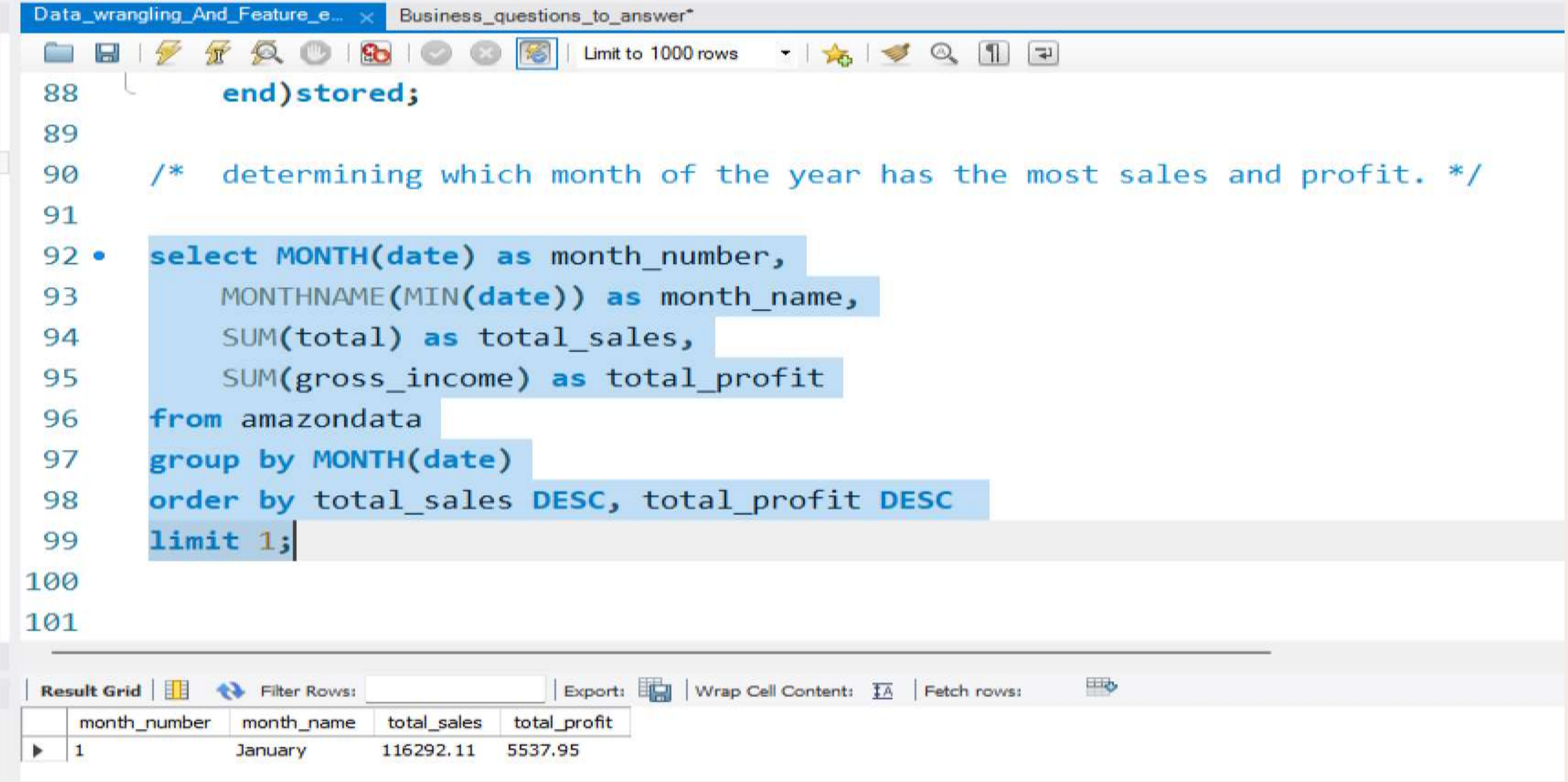
```

/* FEATURE ENGINEERING */

/* Add a new column named timeofday */
alter table amazondata add column  timeofday VARCHAR(15)
as (
    case
        when TIME(time) between '00:00:00' AND '11:59:59' then 'Morning'
        when TIME(time) between '12:00:00' AND '17:59:59' then 'Afternoon'
        when TIME(time) between '18:00:00' AND '23:59:59' then 'Evening'
        else 'Unknown'
    end
    )stored;
```



```
/* Add a new column named dayname */

alter table amazondata add column  dayname VARCHAR(3)
as (
    case
        when DAYOFWEEK(date) = 1 then 'Sun'
        when DAYOFWEEK(date) = 2 then 'Mon'
        when DAYOFWEEK(date) = 3 then 'Tue'
        when DAYOFWEEK(date) = 4 then 'Wed'
        when DAYOFWEEK(date) = 5 then 'Thu'
        when DAYOFWEEK(date) = 6 then 'Fri'
        when DAYOFWEEK(date) = 7 THEN 'Sat'
        else 'Unknown'
    end)
    stored;
```


Data_wrangling_And_Feature_e...
Business_questions_to_answer*
Limit to 1000 rows

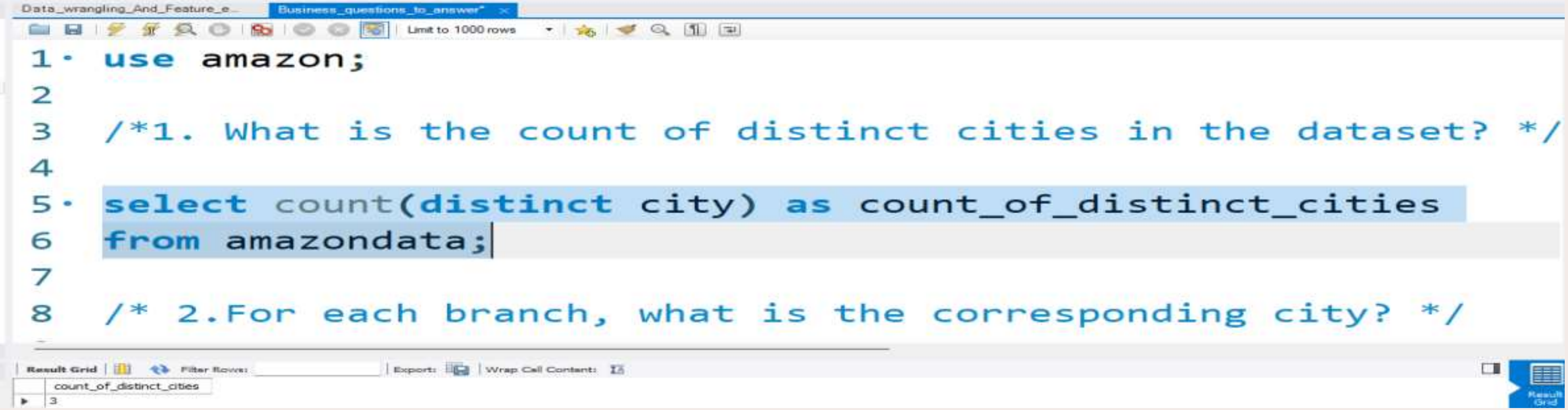
```sql
/* Add a new column named monthname */
alter table amazondata add column  monthname VARCHAR(3)
as (
    case
        when MONTH(date) = 1 then 'Jan'
        when MONTH(date) = 2 then 'Feb'
        when MONTH(date) = 3 then 'Mar'
        when MONTH(date) = 4 then 'Apr'
        when MONTH(date) = 5 then 'May'
        when MONTH(date) = 6 then 'Jun'
        when MONTH(date) = 7 then 'Jul'
        when MONTH(date) = 8 then 'Aug'
        when MONTH(date) = 9 then 'Sep'
        when MONTH(date) = 10 then 'Oct'
        when MONTH(date) = 11 then 'Nov'
        when MONTH(date) = 12 then 'Dec'
        else 'Unknown'
    end)stored;
```



```
        end)stored;

/*  determining which month of the year has the most sales and profit. */

select MONTH(date) as month_number,
    MONTHNAME(MIN(date)) as month_name,
    SUM(total) as total_sales,
    SUM(gross_income) as total_profit
from amazondata
group by MONTH(date)
order by total_sales DESC, total_profit DESC
limit 1;
```

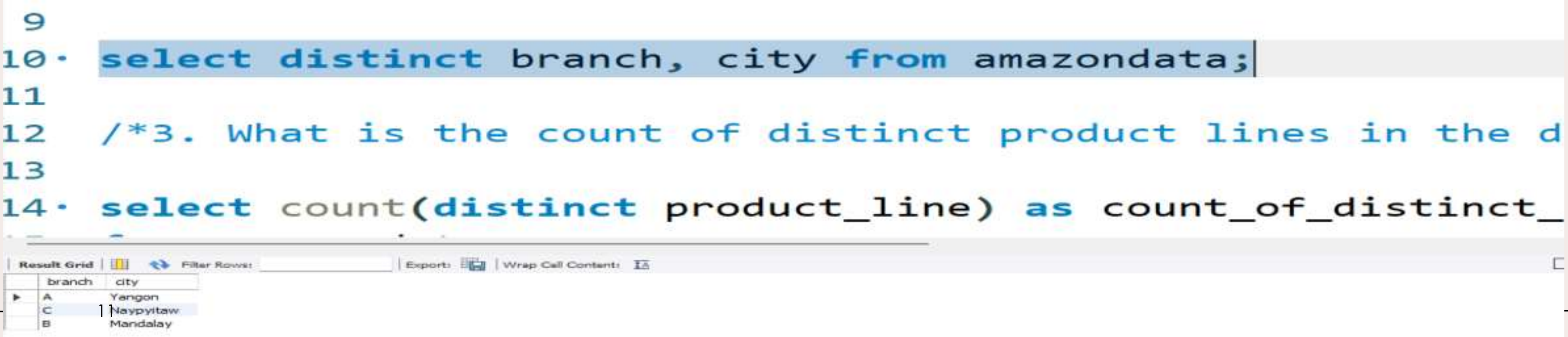
month_number	month_name	total_sales	total_profit
1	January	116292.11	5537.95

Business Questions to answer:



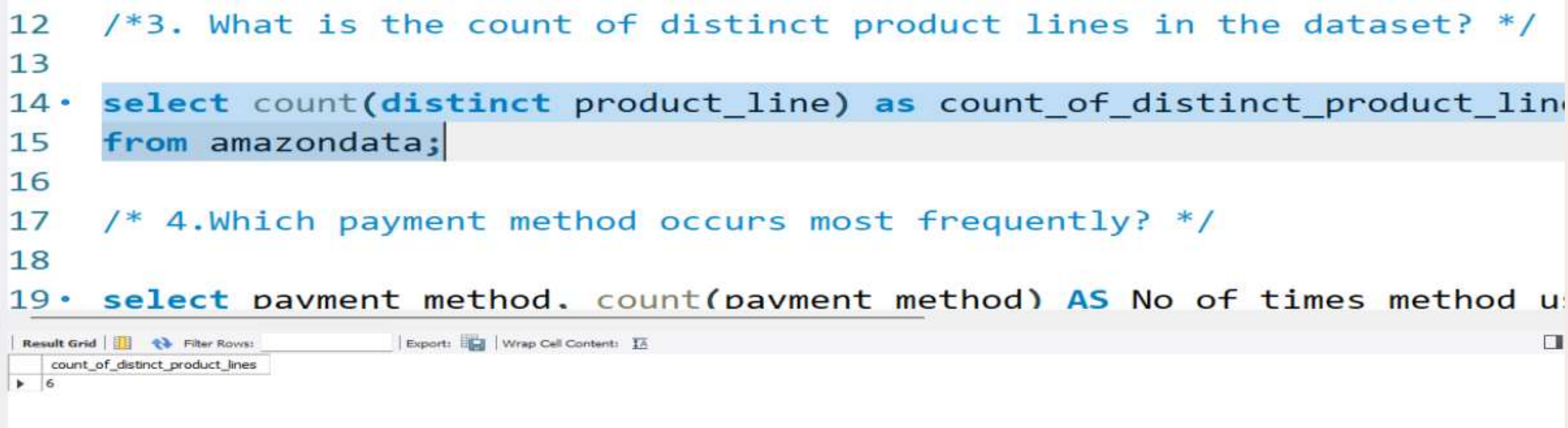
```
use amazon;

/*1. What is the count of distinct cities in the dataset? */

select count(distinct city) as count_of_distinct_cities
from amazondata;

/* 2.For each branch, what is the corresponding city? */
```

```
count_of_distinct_cities
3
```



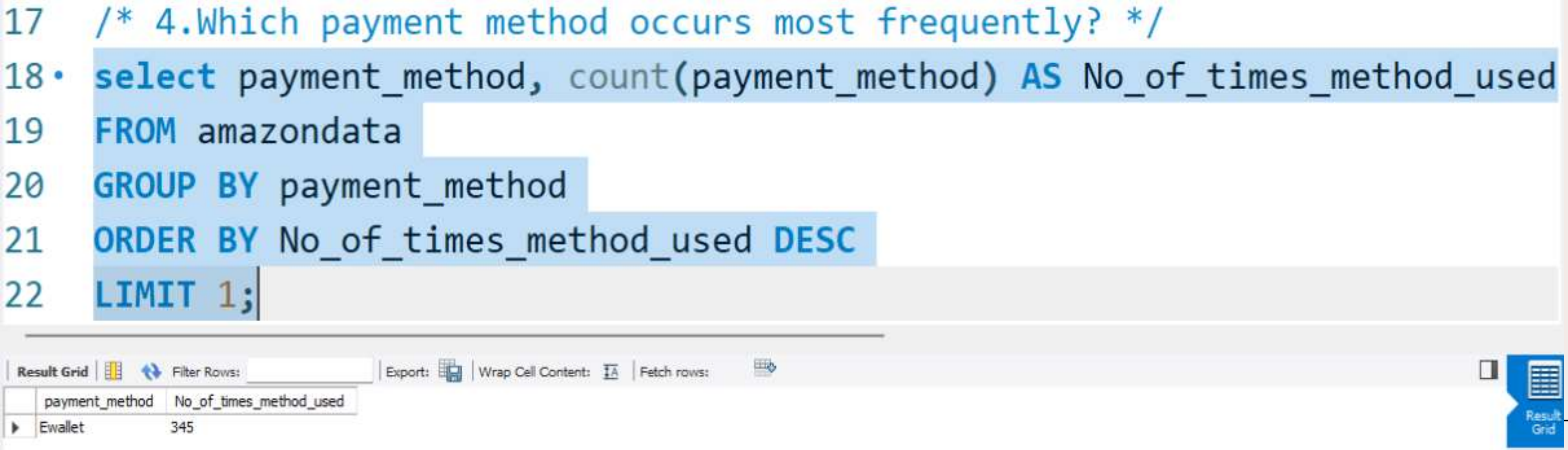
```

select distinct branch, city from amazondata;

/*3. What is the count of distinct product lines in the d

select count(distinct product_line) as count_of_distinct_
```

```
branch  city
A       Yangon
C       Naypyitaw
B       Mandalay
```



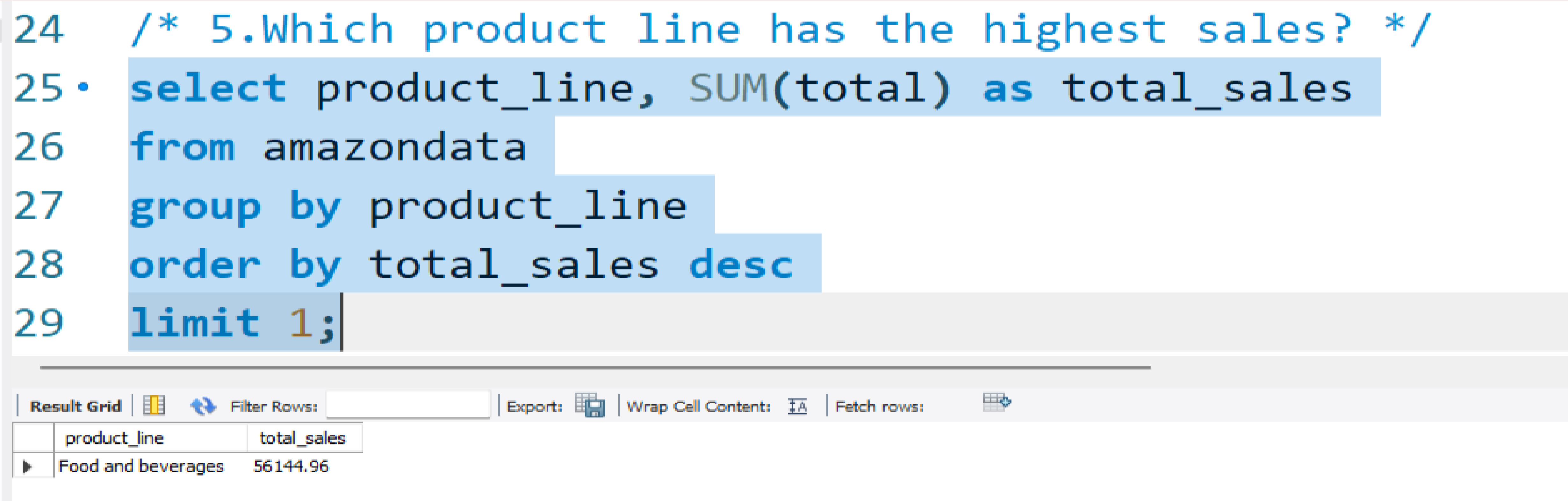
```
/*3. What is the count of distinct product lines in the dataset? */

select count(distinct product_line) as count_of_distinct_product_lin
from amazondata;

/* 4.Which payment method occurs most frequently? */

select payment_method, count(payment_method) AS No_of_times_method_u
```

```
count_of_distinct_product_lines
6
```



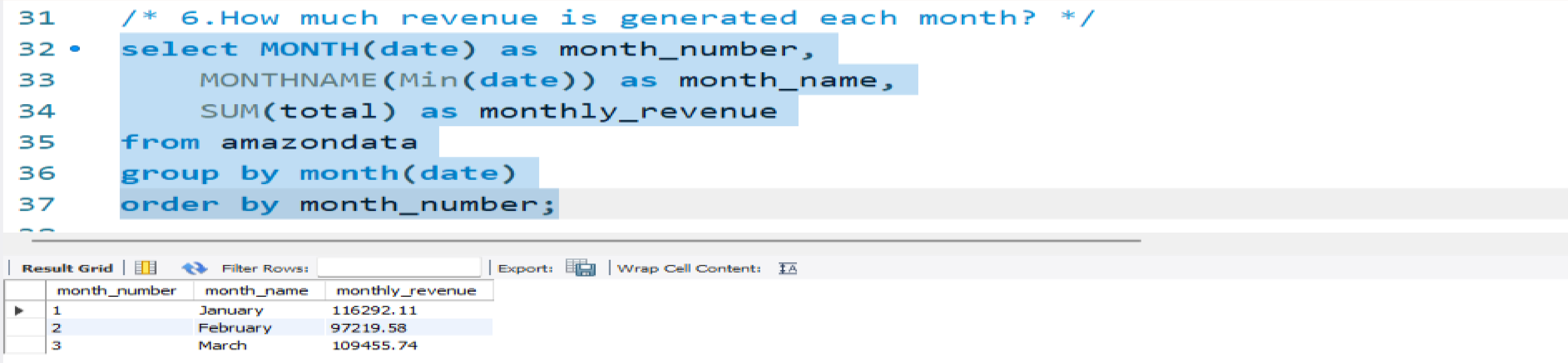
```
/* 4.Which payment method occurs most frequently? */
select payment_method, count(payment_method) AS No_of_times_method_used
FROM amazondata
GROUP BY payment_method
ORDER BY No_of_times_method_used DESC
LIMIT 1;
```

```
payment_method   No_of_times_method_used
Ewallet          345
```



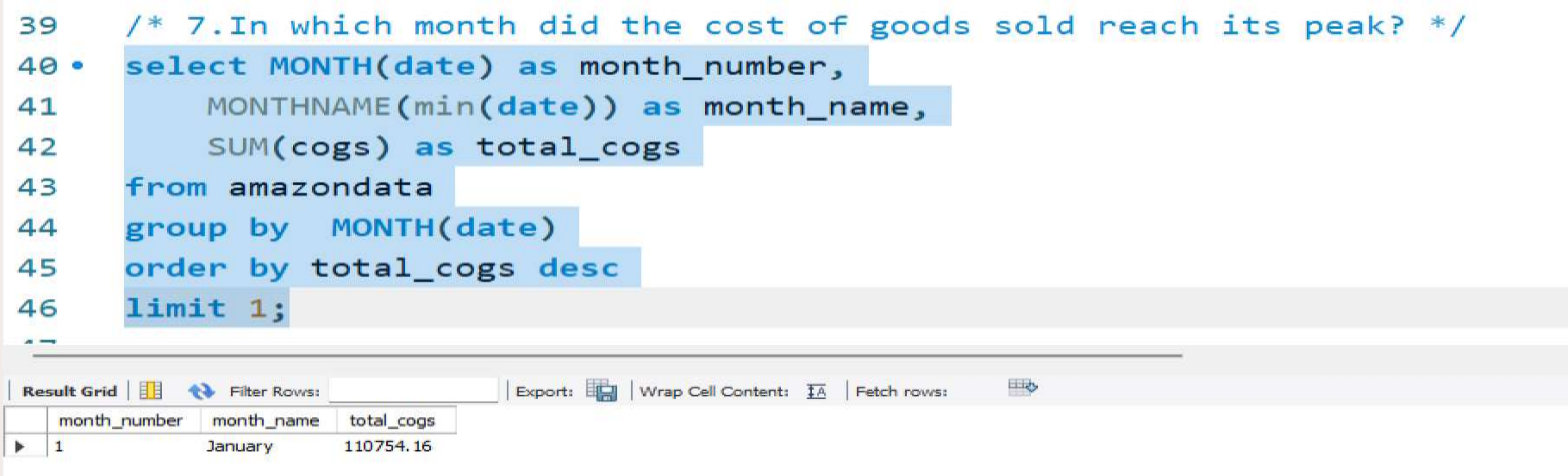
```
/* 5.Which product line has the highest sales? */
select product_line, SUM(total) as total_sales
from amazondata
group by product_line
order by total_sales desc
limit 1;
```

product_line	total_sales
Food and beverages	56144.96



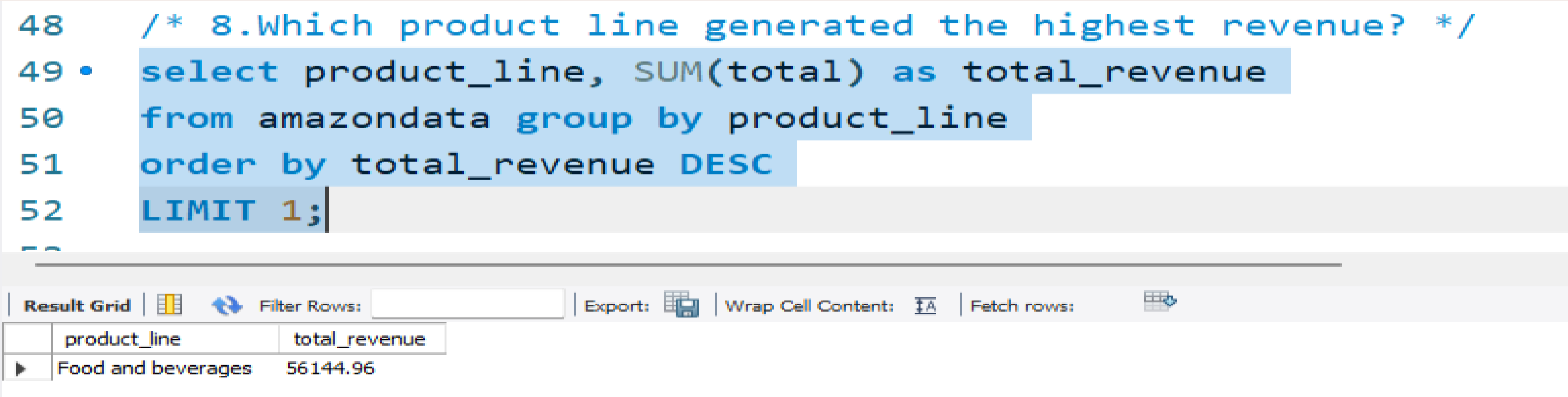
```
/* 6.How much revenue is generated each month? */
select MONTH(date) as month_number,
    MONTHNAME(Min(date)) as month_name,
    SUM(total) as monthly_revenue
from amazondata
group by month(date)
order by month_number;
```

month_number	month_name	monthly_revenue
1	January	116292.11
2	February	97219.58
3	March	109455.74



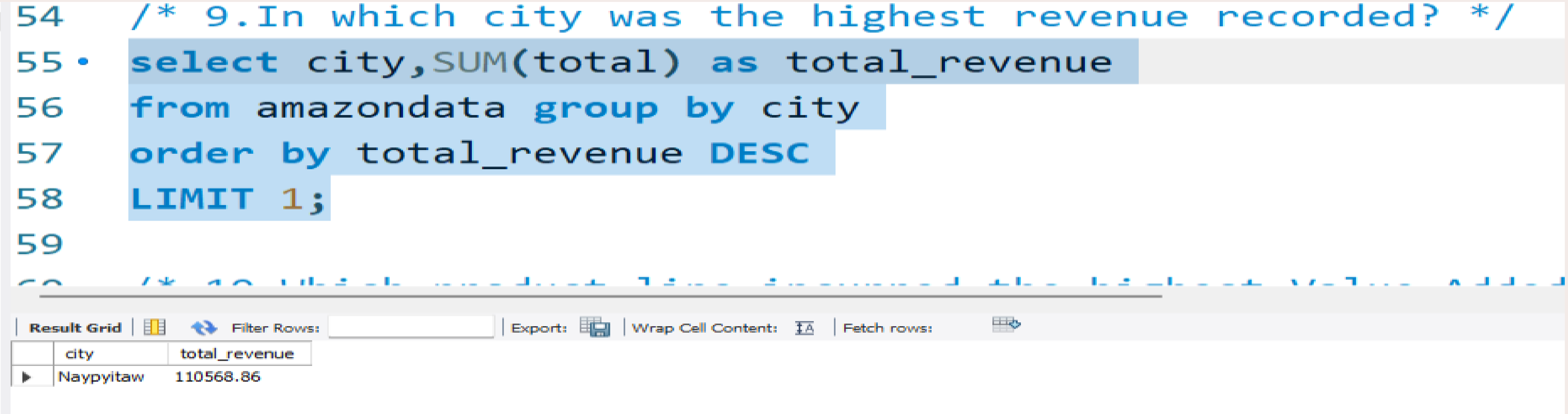
```
/* 7.In which month did the cost of goods sold reach its peak? */
select MONTH(date) as month_number,
    MONTHNAME(min(date)) as month_name,
    SUM(cogs) as total_cogs
from amazondata
group by  MONTH(date)
order by total_cogs desc
limit 1;
```

```
month_number   month_name   total_cogs
1              January      110754.16
```



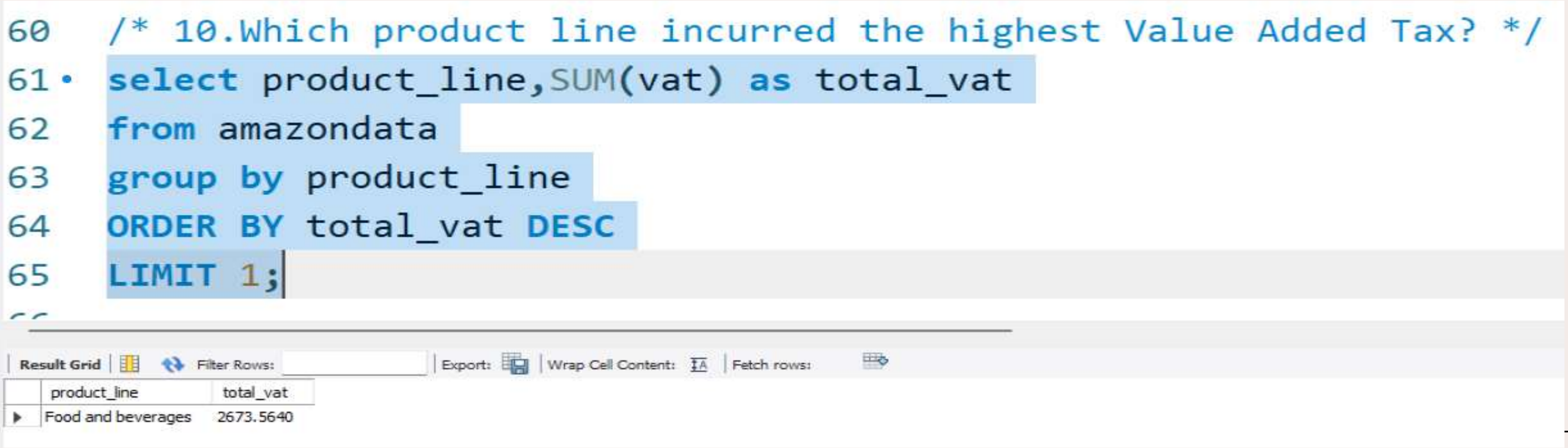
```
/* 8.Which product line generated the highest revenue? */
select product_line, SUM(total) as total_revenue
from amazondata group by product_line
order by total_revenue DESC
LIMIT 1;
```

```
product_line          total_revenue
Food and beverages    56144.96
```



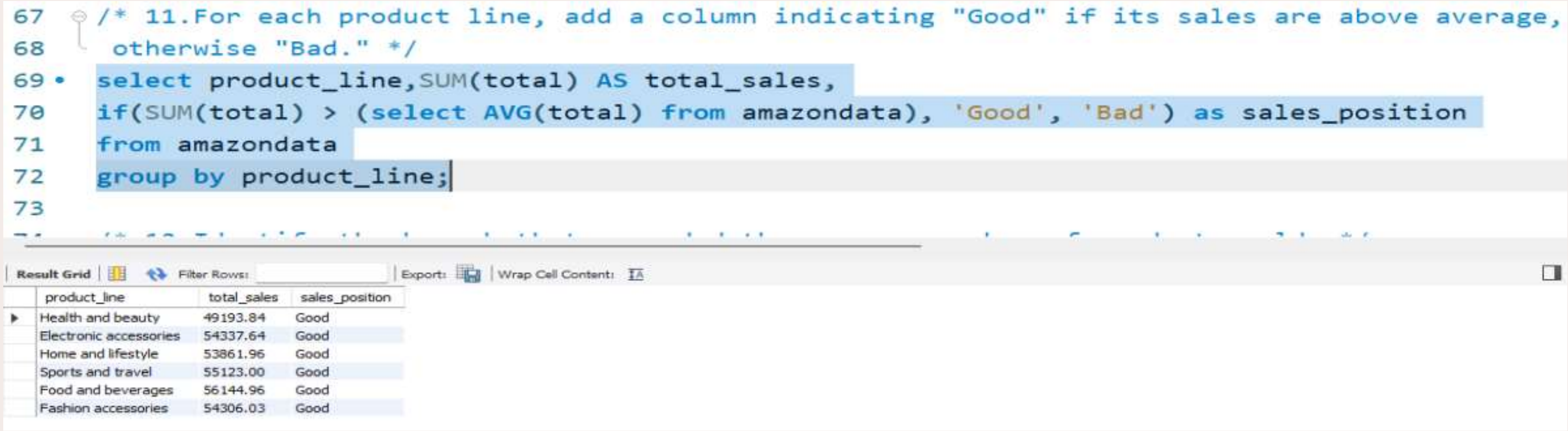
```sql
/* 9.In which city was the highest revenue recorded? */
select city,SUM(total) as total_revenue
from amazondata group by city
order by total_revenue DESC
LIMIT 1;
```

```
   city        total_revenue
▶  Naypyitaw   110568.86
```



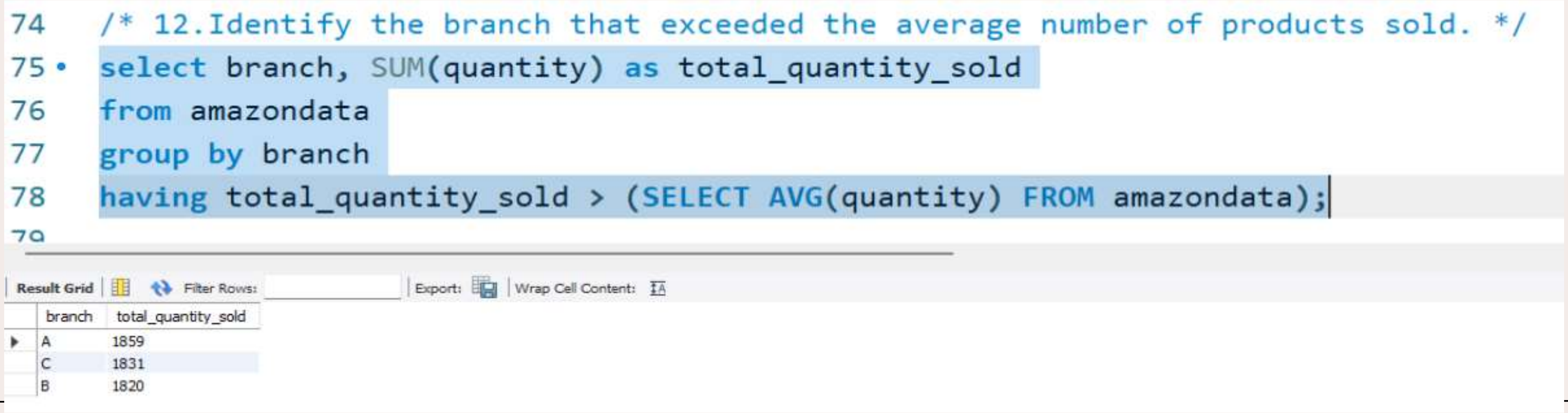
```sql
/* 10.Which product line incurred the highest Value Added Tax? */
select product_line,SUM(vat) as total_vat
from amazondata
group by product_line
ORDER BY total_vat DESC
LIMIT 1;
```

```
   product_line         total_vat
▶  Food and beverages   2673.5640
```



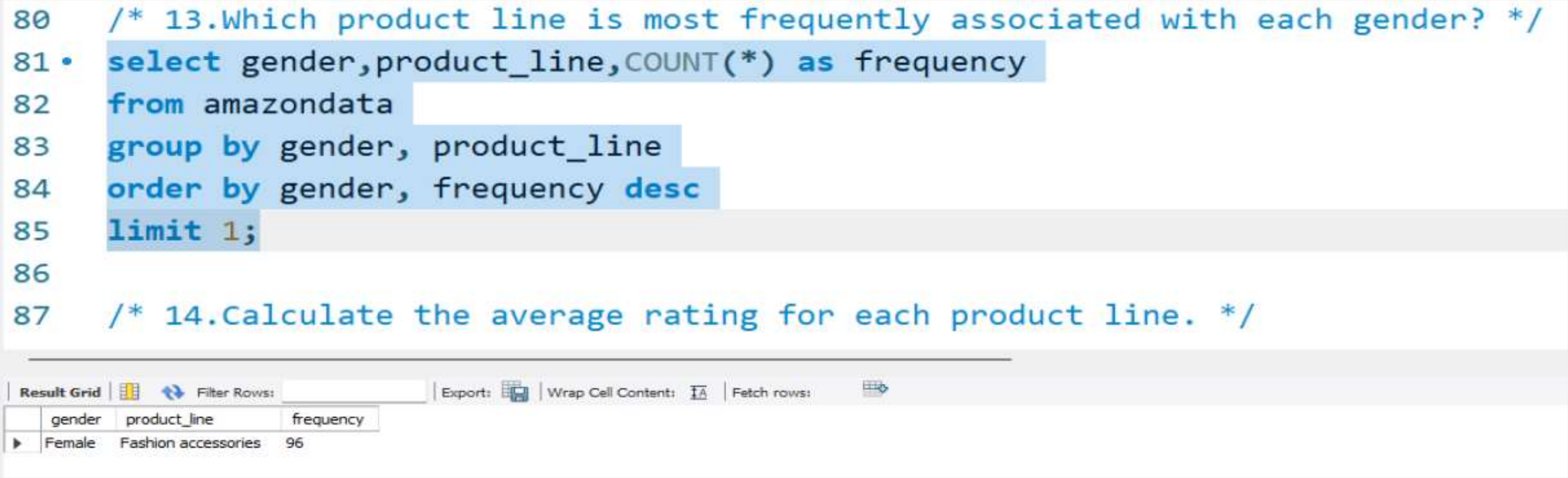
```sql
/* 11.For each product line, add a column indicating "Good" if its sales are above average,
 otherwise "Bad." */
select product_line,SUM(total) AS total_sales,
if(SUM(total) > (select AVG(total) from amazondata), 'Good', 'Bad') as sales_position
from amazondata
group by product_line;
```

product_line	total_sales	sales_position
Health and beauty	49193.84	Good
Electronic accessories	54337.64	Good
Home and lifestyle	53861.96	Good
Sports and travel	55123.00	Good
Food and beverages	56144.96	Good
Fashion accessories	54306.03	Good



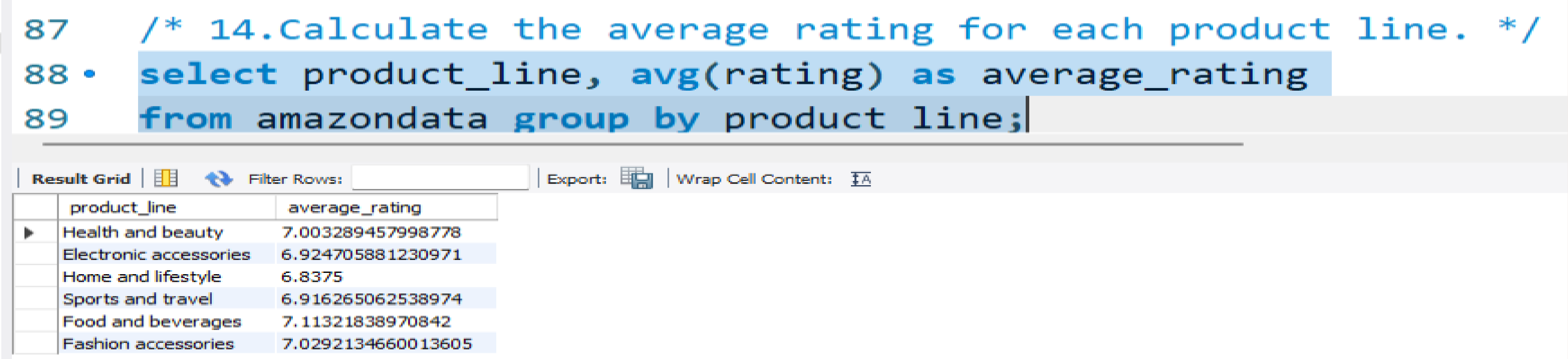
```sql
/* 12.Identify the branch that exceeded the average number of products sold. */
select branch, SUM(quantity) as total_quantity_sold
from amazondata
group by branch
having total_quantity_sold > (SELECT AVG(quantity) FROM amazondata);
```

branch	total_quantity_sold
A	1859
C	1831
B	1820



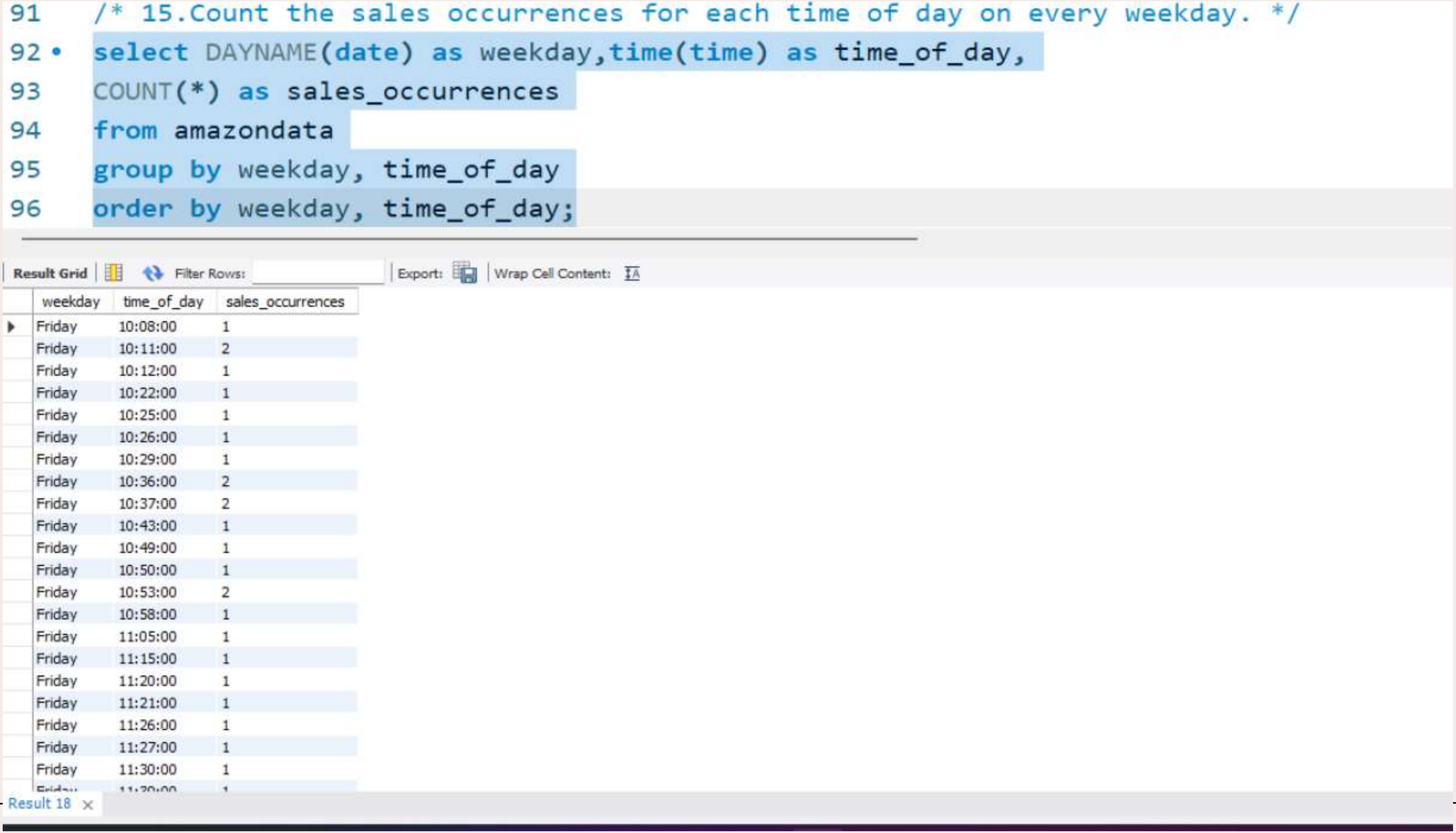
```sql
/* 13.Which product line is most frequently associated with each gender? */
select gender,product_line,COUNT(*) as frequency
from amazondata
group by gender, product_line
order by gender, frequency desc
limit 1;

/* 14.Calculate the average rating for each product line. */
```

gender	product_line	frequency
Female	Fashion accessories	96



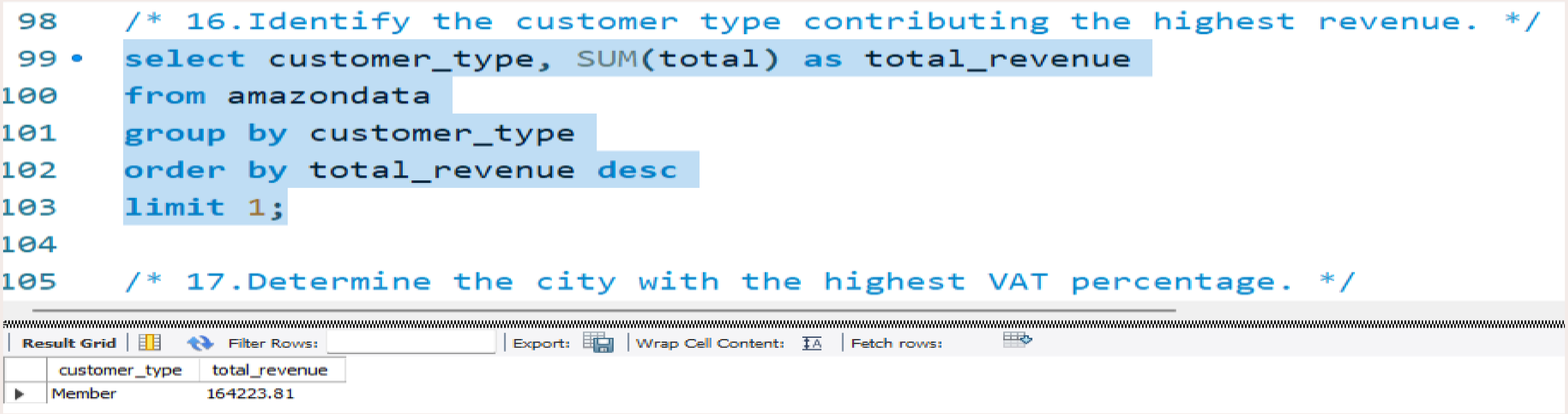
```sql
/* 14.Calculate the average rating for each product line. */
select product_line, avg(rating) as average_rating
from amazondata group by product_line;
```

product_line	average_rating
Health and beauty	7.003289457998778
Electronic accessories	6.924705881230971
Home and lifestyle	6.8375
Sports and travel	6.916265062538974
Food and beverages	7.11321838970842
Fashion accessories	7.029213460013605



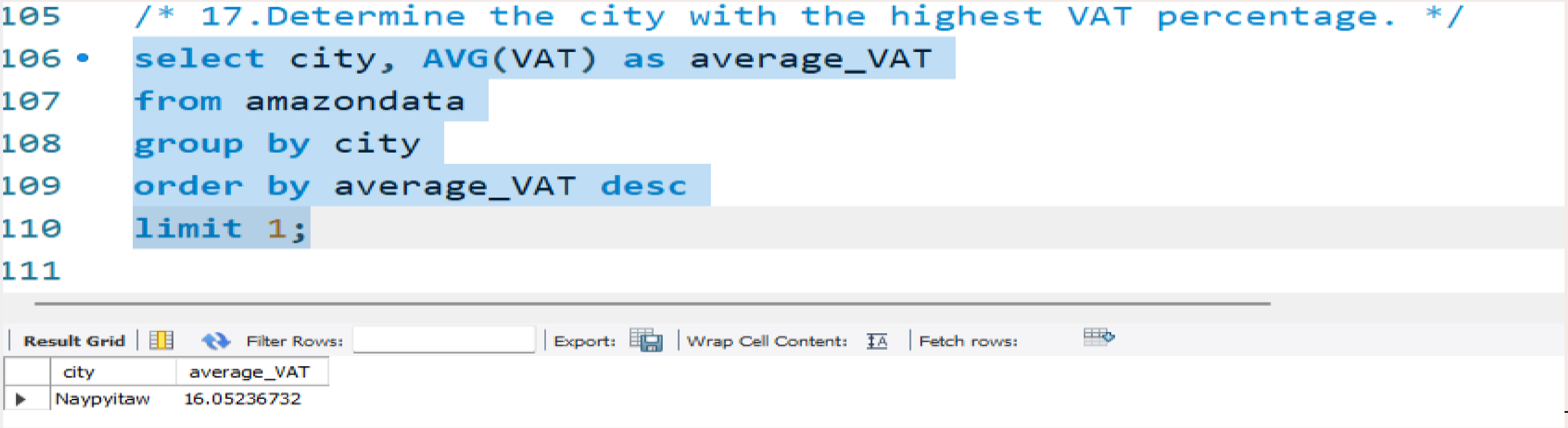
```sql
/* 15.Count the sales occurrences for each time of day on every weekday. */
select DAYNAME(date) as weekday,time(time) as time_of_day,
COUNT(*) as sales_occurrences
from amazondata
group by weekday, time_of_day
order by weekday, time_of_day;
```

weekday	time_of_day	sales_occurrences
Friday	10:08:00	1
Friday	10:11:00	2
Friday	10:12:00	1
Friday	10:22:00	1
Friday	10:25:00	1
Friday	10:26:00	1
Friday	10:29:00	1
Friday	10:36:00	2
Friday	10:37:00	2
Friday	10:43:00	1
Friday	10:49:00	1
Friday	10:50:00	1
Friday	10:53:00	2
Friday	10:58:00	1
Friday	11:05:00	1
Friday	11:15:00	1
Friday	11:20:00	1
Friday	11:21:00	1
Friday	11:26:00	1
Friday	11:27:00	1
Friday	11:30:00	1



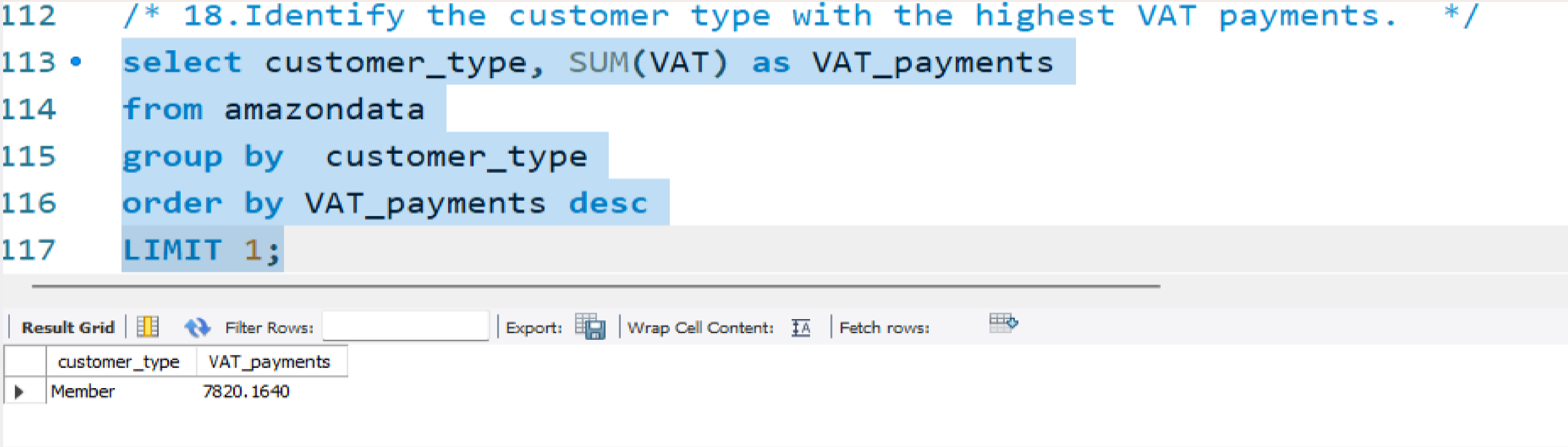
```
/* 16.Identify the customer type contributing the highest revenue. */
select customer_type, SUM(total) as total_revenue
from amazondata
group by customer_type
order by total_revenue desc
limit 1;

/* 17.Determine the city with the highest VAT percentage. */
```

	customer_type	total_revenue
▶	Member	164223.81



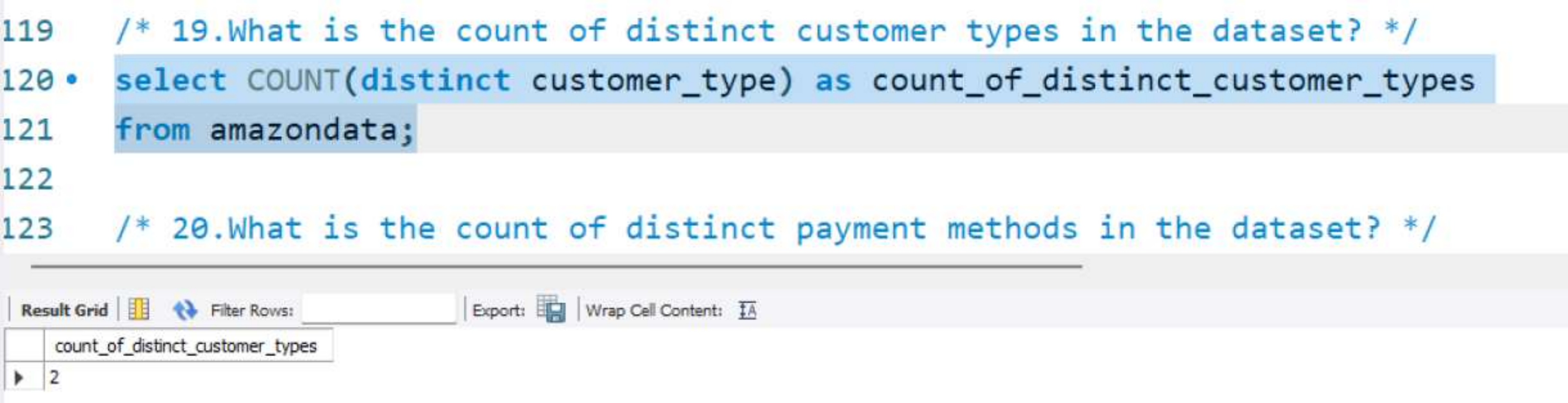
```
/* 17.Determine the city with the highest VAT percentage. */
select city, AVG(VAT) as average_VAT
from amazondata
group by city
order by average_VAT desc
limit 1;
```

	city	average_VAT
▶	Naypyitaw	16.05236732



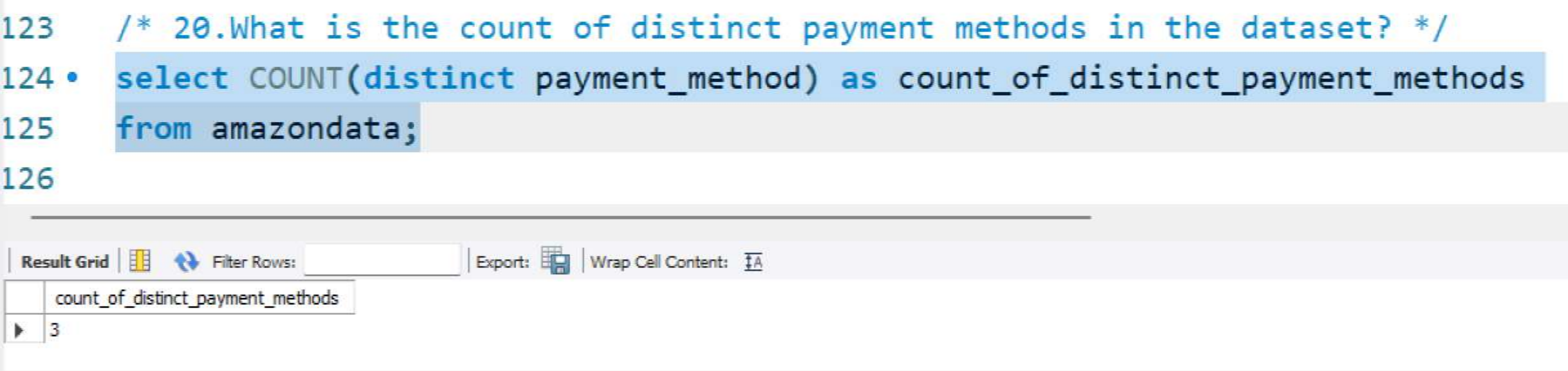
```sql
/* 18.Identify the customer type with the highest VAT payments.  */
select customer_type, SUM(VAT) as VAT_payments
from amazondata
group by  customer_type
order by VAT_payments desc
LIMIT 1;
```

customer_type	VAT_payments
Member	7820.1640



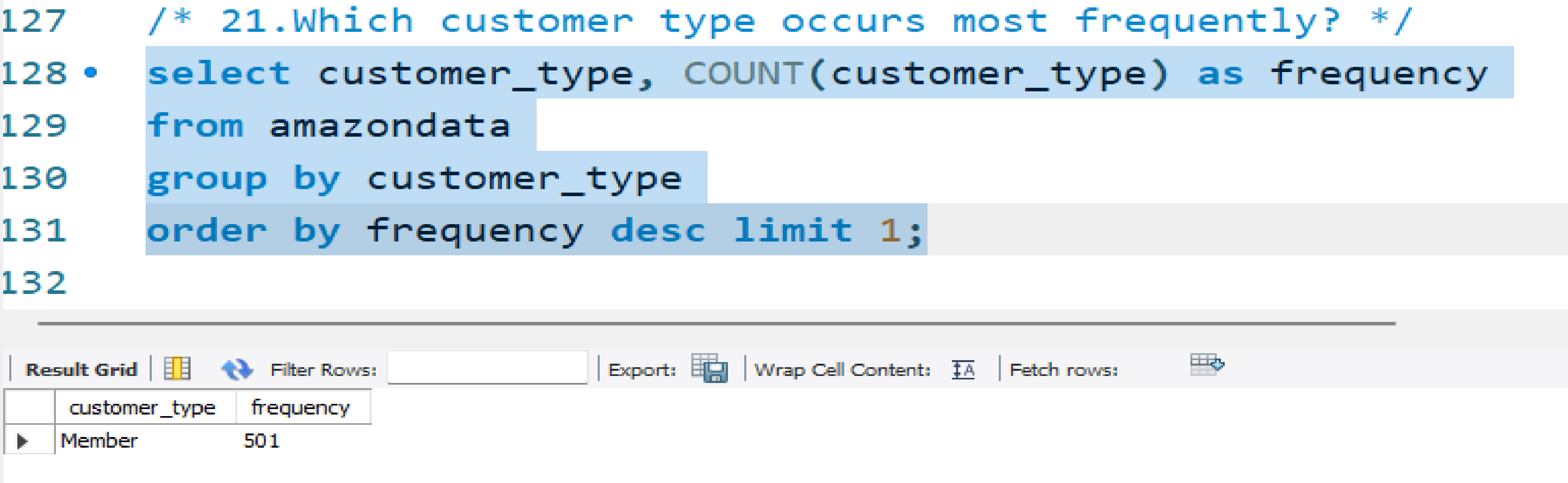
```sql
/* 19.What is the count of distinct customer types in the dataset? */
select COUNT(distinct customer_type) as count_of_distinct_customer_types
from amazondata;

/* 20.What is the count of distinct payment methods in the dataset? */
```

count_of_distinct_customer_types
2



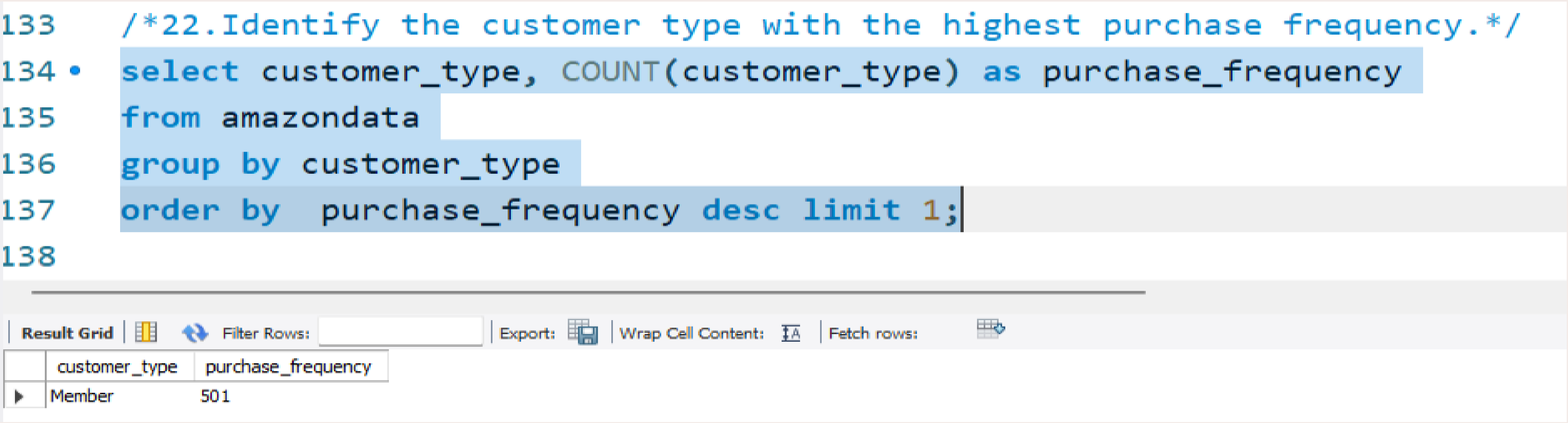
```sql
/* 20.What is the count of distinct payment methods in the dataset? */
select COUNT(distinct payment_method) as count_of_distinct_payment_methods
from amazondata;
```

count_of_distinct_payment_methods
3



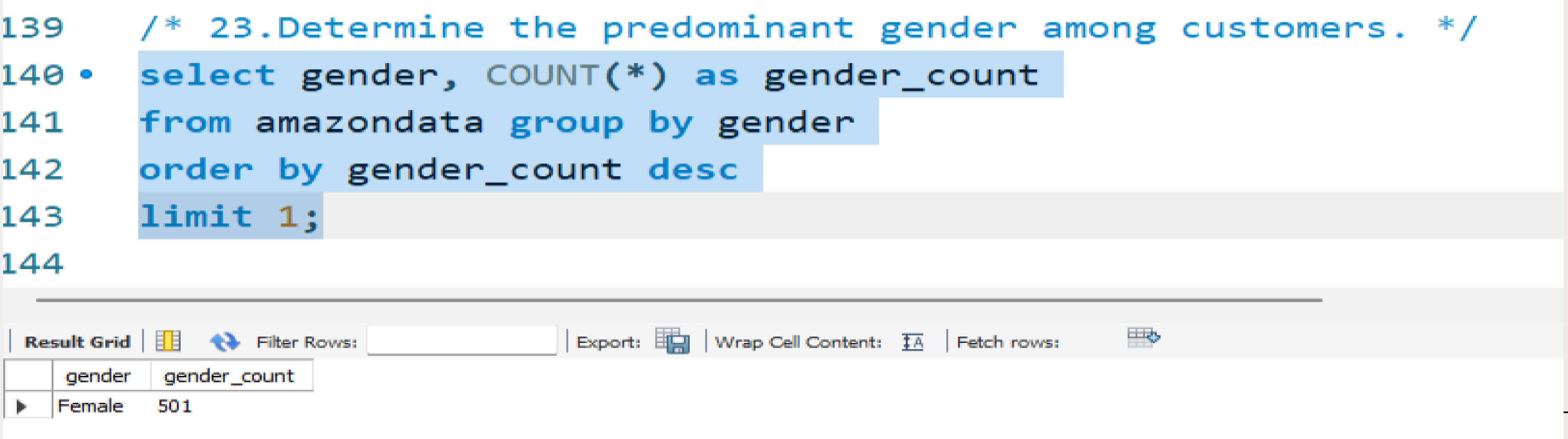
```sql
/* 21.Which customer type occurs most frequently? */
select customer_type, COUNT(customer_type) as frequency
from amazondata
group by customer_type
order by frequency desc limit 1;
```

customer_type	frequency
Member	501



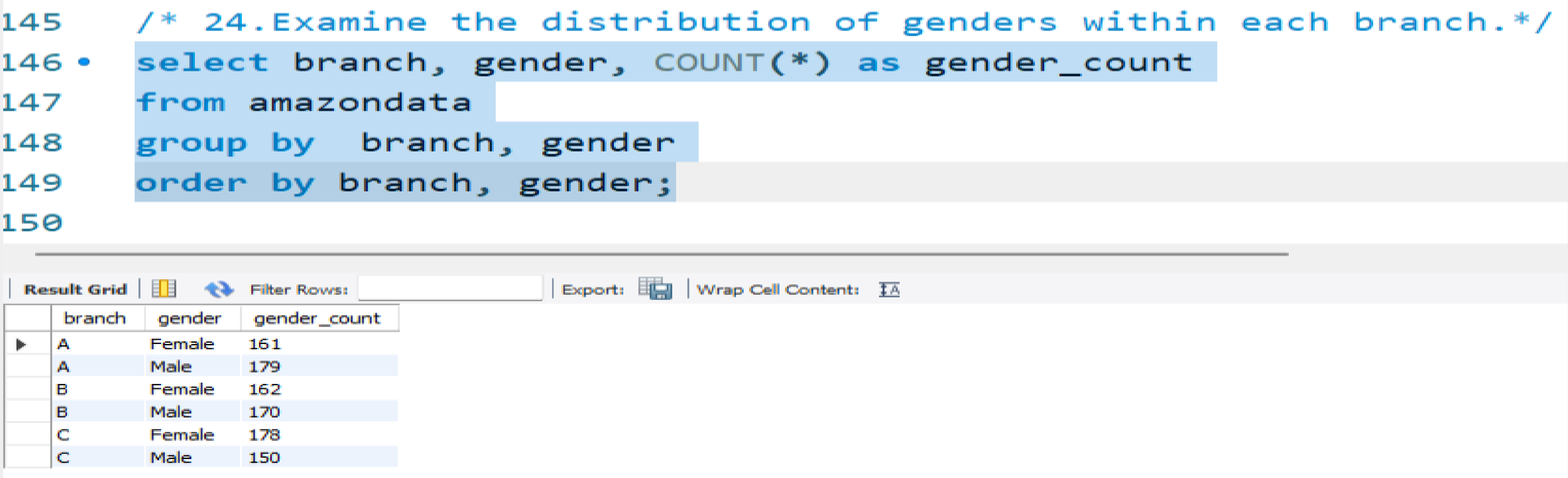
```sql
/*22.Identify the customer type with the highest purchase frequency.*/
select customer_type, COUNT(customer_type) as purchase_frequency
from amazondata
group by customer_type
order by  purchase_frequency desc limit 1;
```

customer_type	purchase_frequency
Member	501



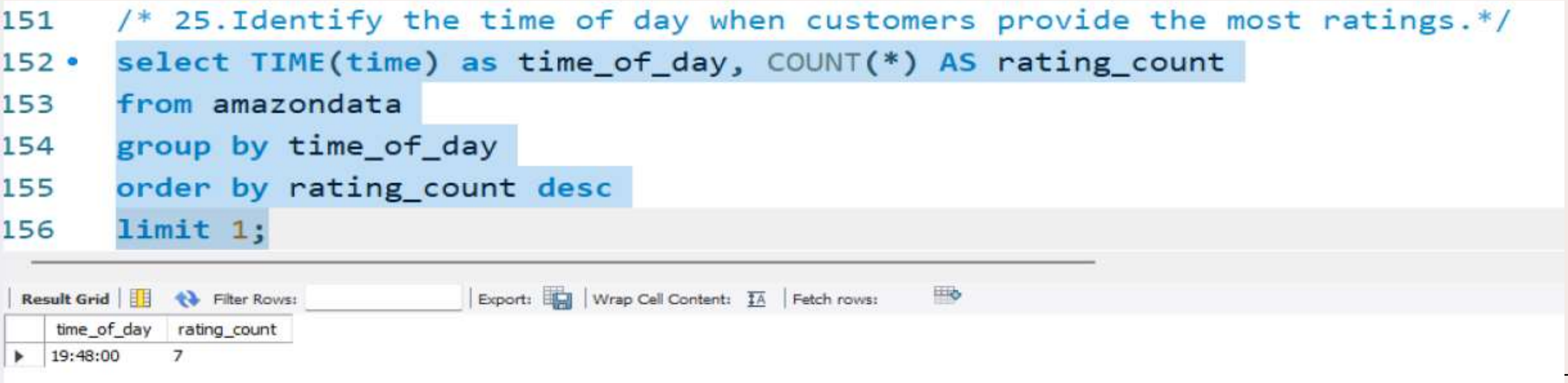
```sql
/* 23.Determine the predominant gender among customers. */
select gender, COUNT(*) as gender_count
from amazondata group by gender
order by gender_count desc
limit 1;
```

gender	gender_count
Female	501



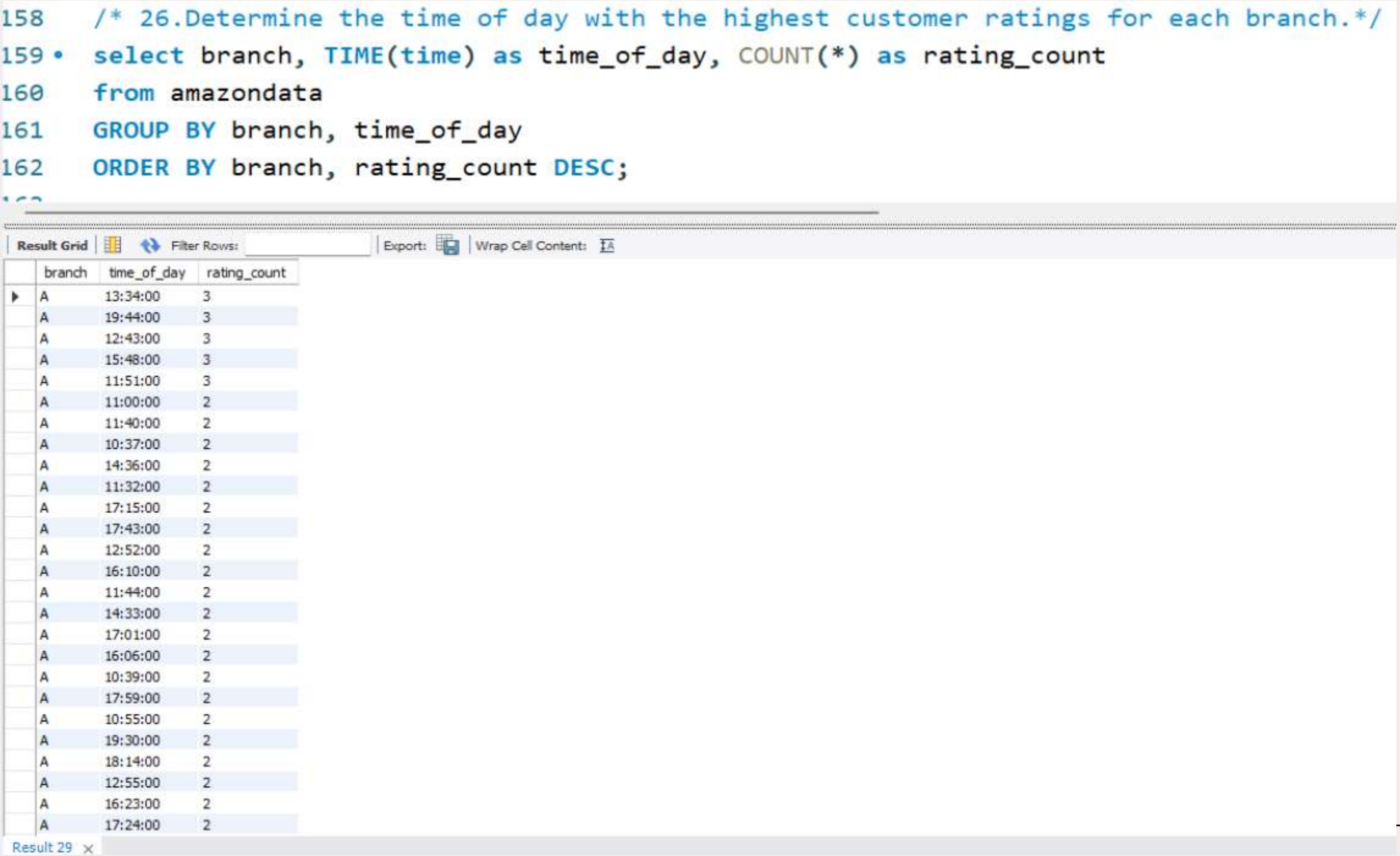
```sql
/* 24.Examine the distribution of genders within each branch.*/
select branch, gender, COUNT(*) as gender_count
from amazondata
group by  branch, gender
order by branch, gender;
```

branch	gender	gender_count
A	Female	161
A	Male	179
B	Female	162
B	Male	170
C	Female	178
C	Male	150



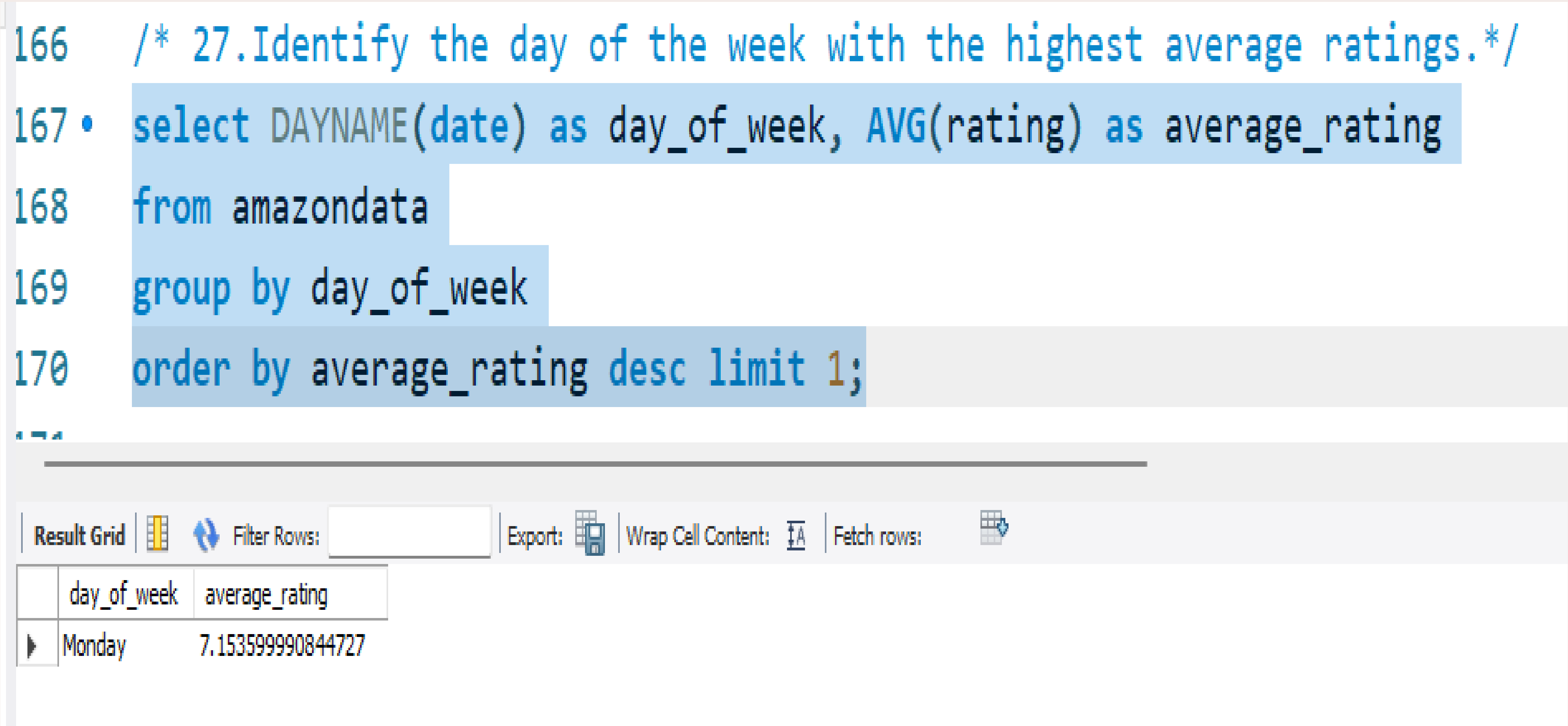
```sql
/* 25.Identify the time of day when customers provide the most ratings.*/
select TIME(time) as time_of_day, COUNT(*) AS rating_count
from amazondata
group by time_of_day
order by rating_count desc
limit 1;
```

time_of_day	rating_count
19:48:00	7



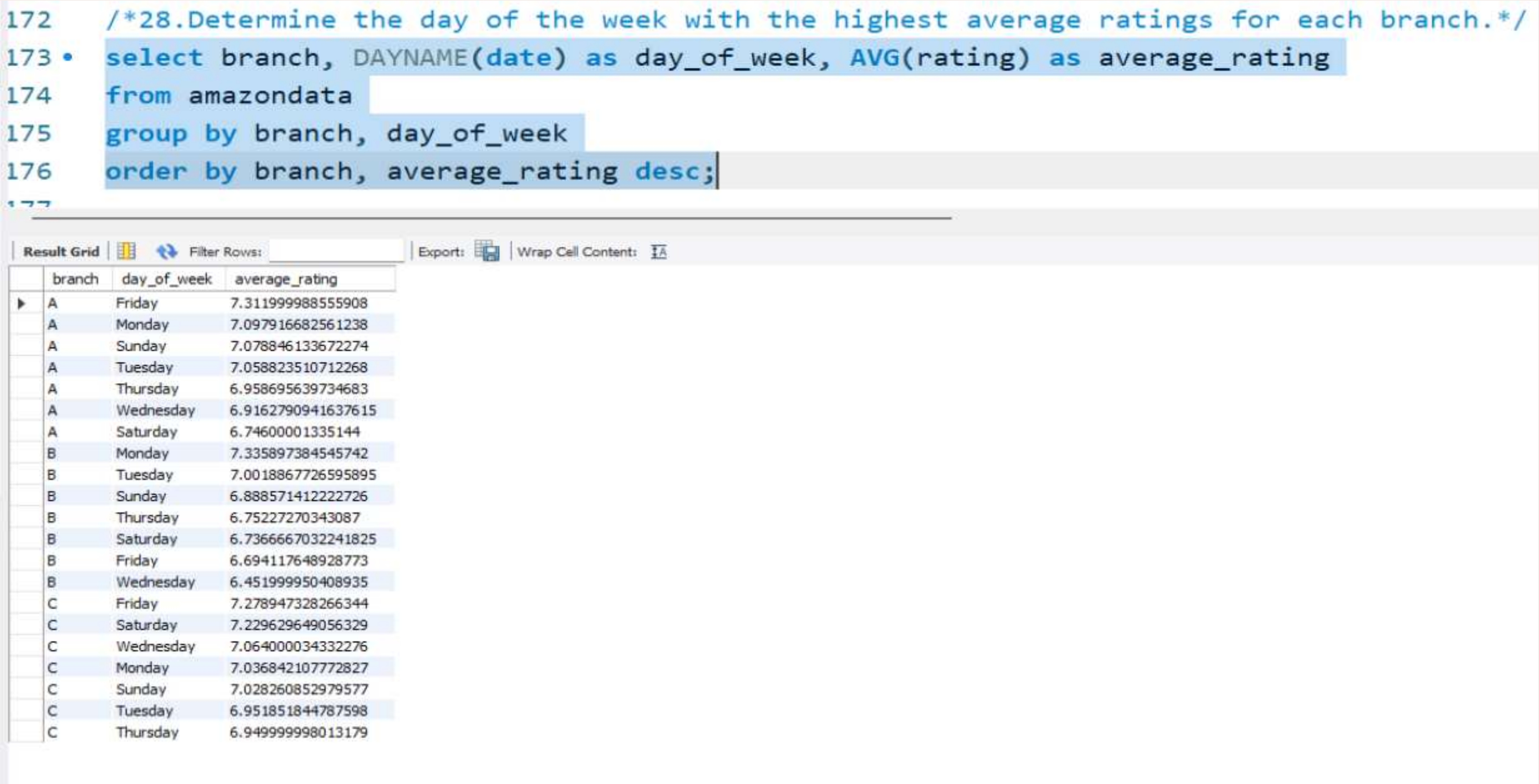
```sql
/* 26.Determine the time of day with the highest customer ratings for each branch.*/
select branch, TIME(time) as time_of_day, COUNT(*) as rating_count
from amazondata
GROUP BY branch, time_of_day
ORDER BY branch, rating_count DESC;
```

branch	time_of_day	rating_count
A	13:34:00	3
A	19:44:00	3
A	12:43:00	3
A	15:48:00	3
A	11:51:00	3
A	11:00:00	2
A	11:40:00	2
A	10:37:00	2
A	14:36:00	2
A	11:32:00	2
A	17:15:00	2
A	17:43:00	2
A	12:52:00	2
A	16:10:00	2
A	11:44:00	2
A	14:33:00	2
A	17:01:00	2
A	16:06:00	2
A	10:39:00	2
A	17:59:00	2
A	10:55:00	2
A	19:30:00	2
A	18:14:00	2
A	12:55:00	2
A	16:23:00	2
A	17:24:00	2



```sql
/* 27.Identify the day of the week with the highest average ratings.*/
select DAYNAME(date) as day_of_week, AVG(rating) as average_rating
from amazondata
group by day_of_week
order by average_rating desc limit 1;
```

day_of_week	average_rating
Monday	7.153599990844727



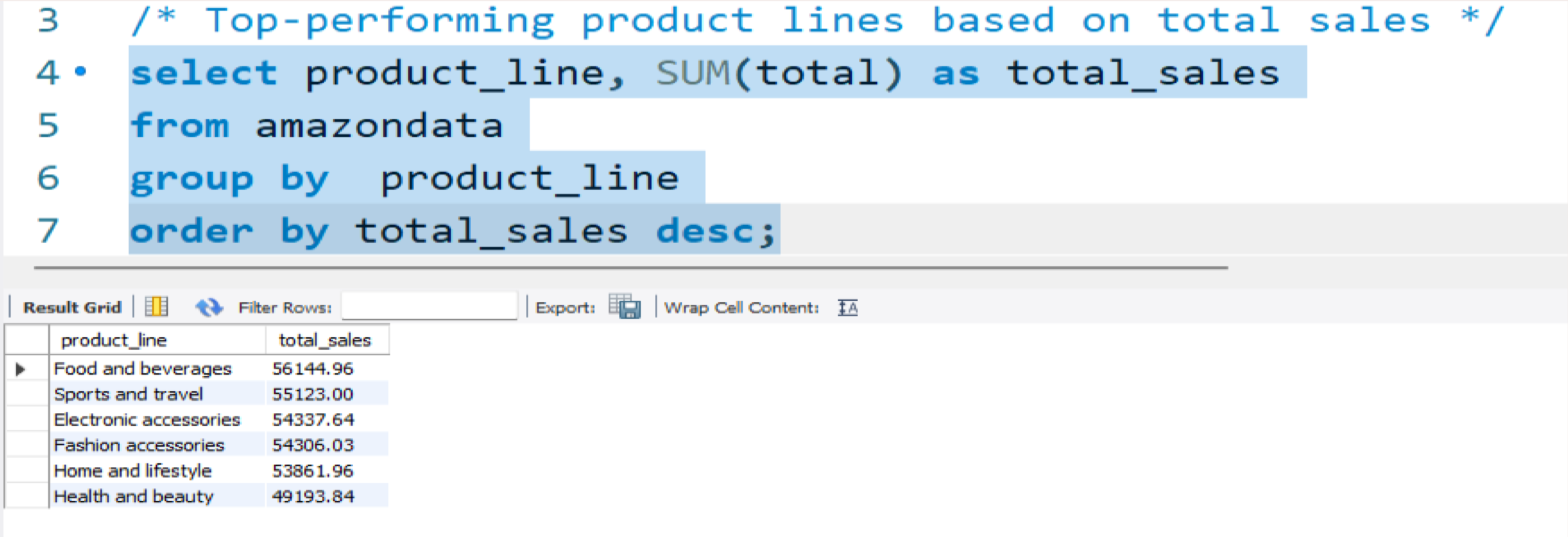
```sql
/*28.Determine the day of the week with the highest average ratings for each branch.*/
select branch, DAYNAME(date) as day_of_week, AVG(rating) as average_rating
from amazondata
group by branch, day_of_week
order by branch, average_rating desc;
```

branch	day_of_week	average_rating
A	Friday	7.311999988555908
A	Monday	7.097916682561238
A	Sunday	7.078846133672274
A	Tuesday	7.058823510712268
A	Thursday	6.958695639734683
A	Wednesday	6.9162790941637615
A	Saturday	6.74600001335144
B	Monday	7.335897384545742
B	Tuesday	7.0018867726595895
B	Sunday	6.888571412222726
B	Thursday	6.75227270343087
B	Saturday	6.7366667032241825
B	Friday	6.694117648928773
B	Wednesday	6.451999950408935
C	Friday	7.278947328266344
C	Saturday	7.229629649056329
C	Wednesday	7.064000034332276
C	Monday	7.036842107772827
C	Sunday	7.028260852979577
C	Tuesday	6.951851844787598
C	Thursday	6.949999998013179

Analysis

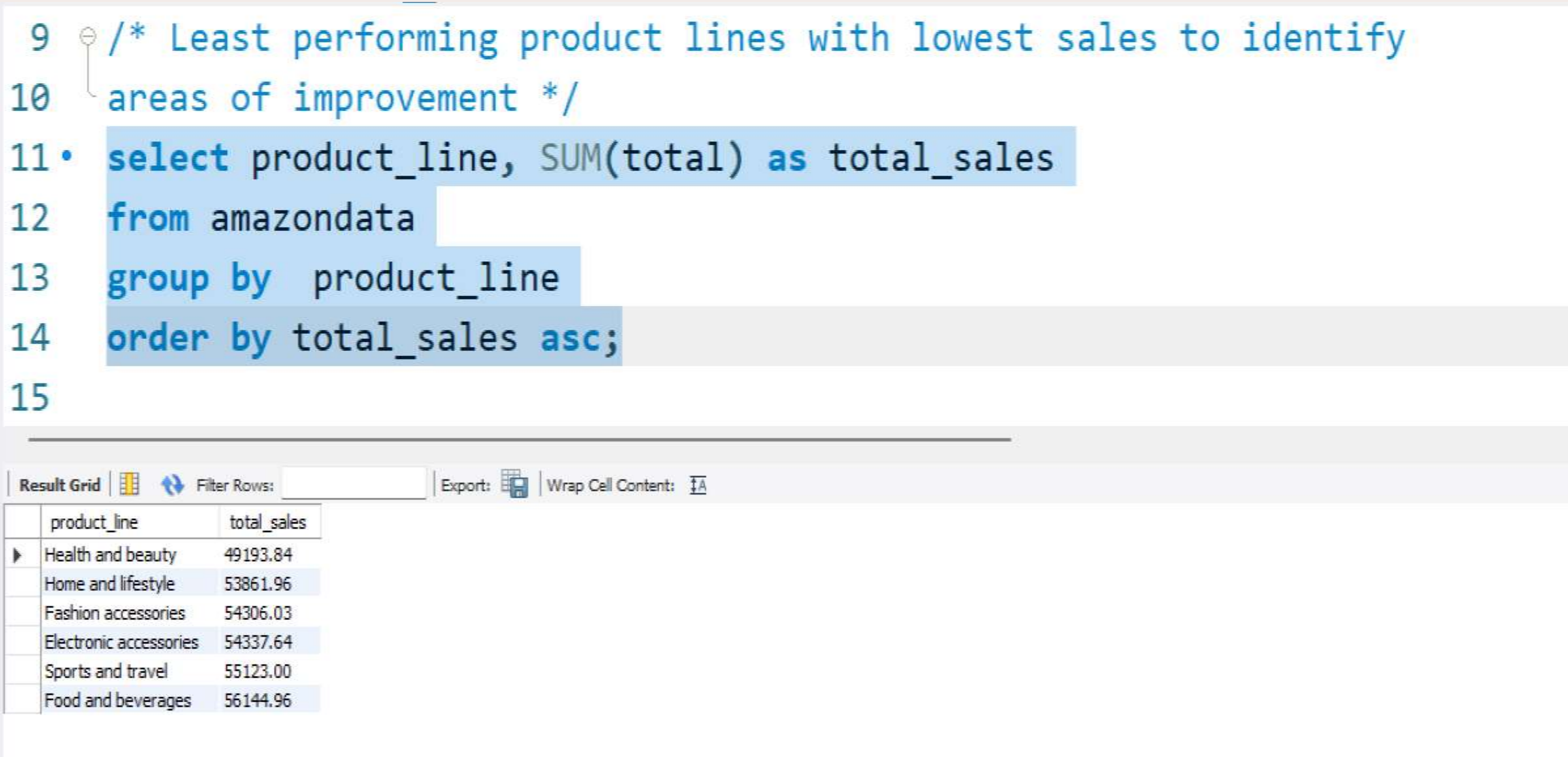
- Product Analysis:

  performance of different product lines, determine which product lines are performing well, and least.



```
/* Top-performing product lines based on total sales */
select product_line, SUM(total) as total_sales
from amazondata
group by  product_line
order by total_sales desc;
```

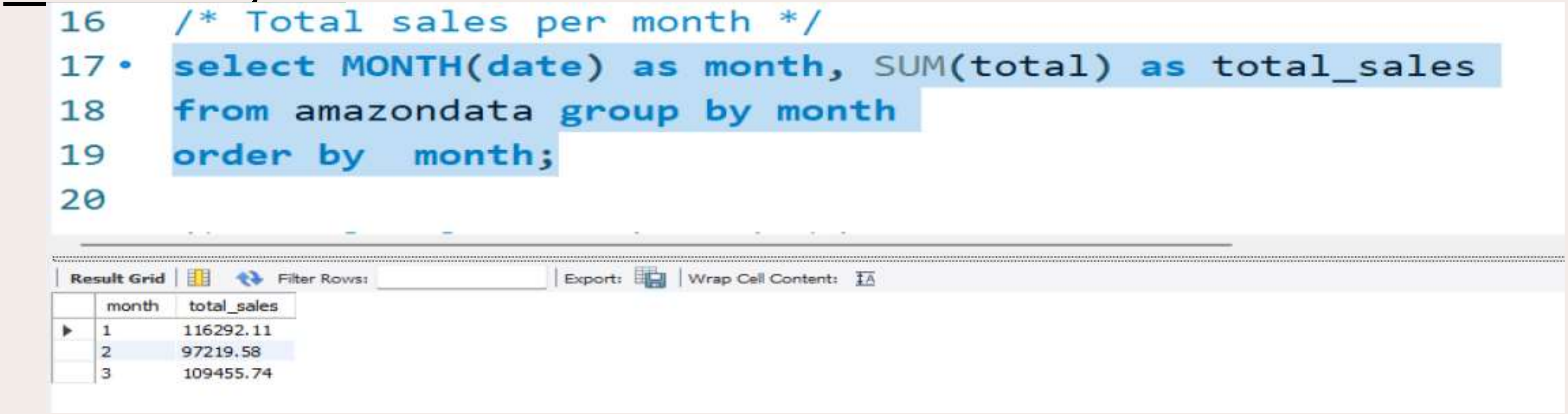
product_line	total_sales
Food and beverages	56144.96
Sports and travel	55123.00
Electronic accessories	54337.64
Fashion accessories	54306.03
Home and lifestyle	53861.96
Health and beauty	49193.84



```
/* Least performing product lines with lowest sales to identify
areas of improvement */
select product_line, SUM(total) as total_sales
from amazondata
group by  product_line
order by total_sales asc;
```

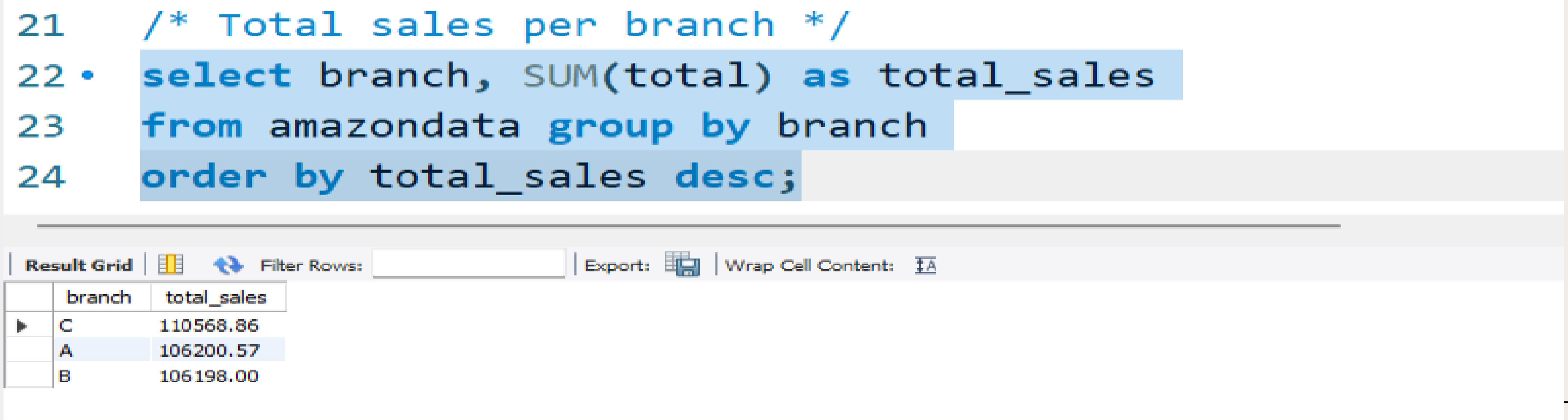
product_line	total_sales
Health and beauty	49193.84
Home and lifestyle	53861.96
Fashion accessories	54306.03
Electronic accessories	54337.64
Sports and travel	55123.00
Food and beverages	56144.96

- Sales Analysis:



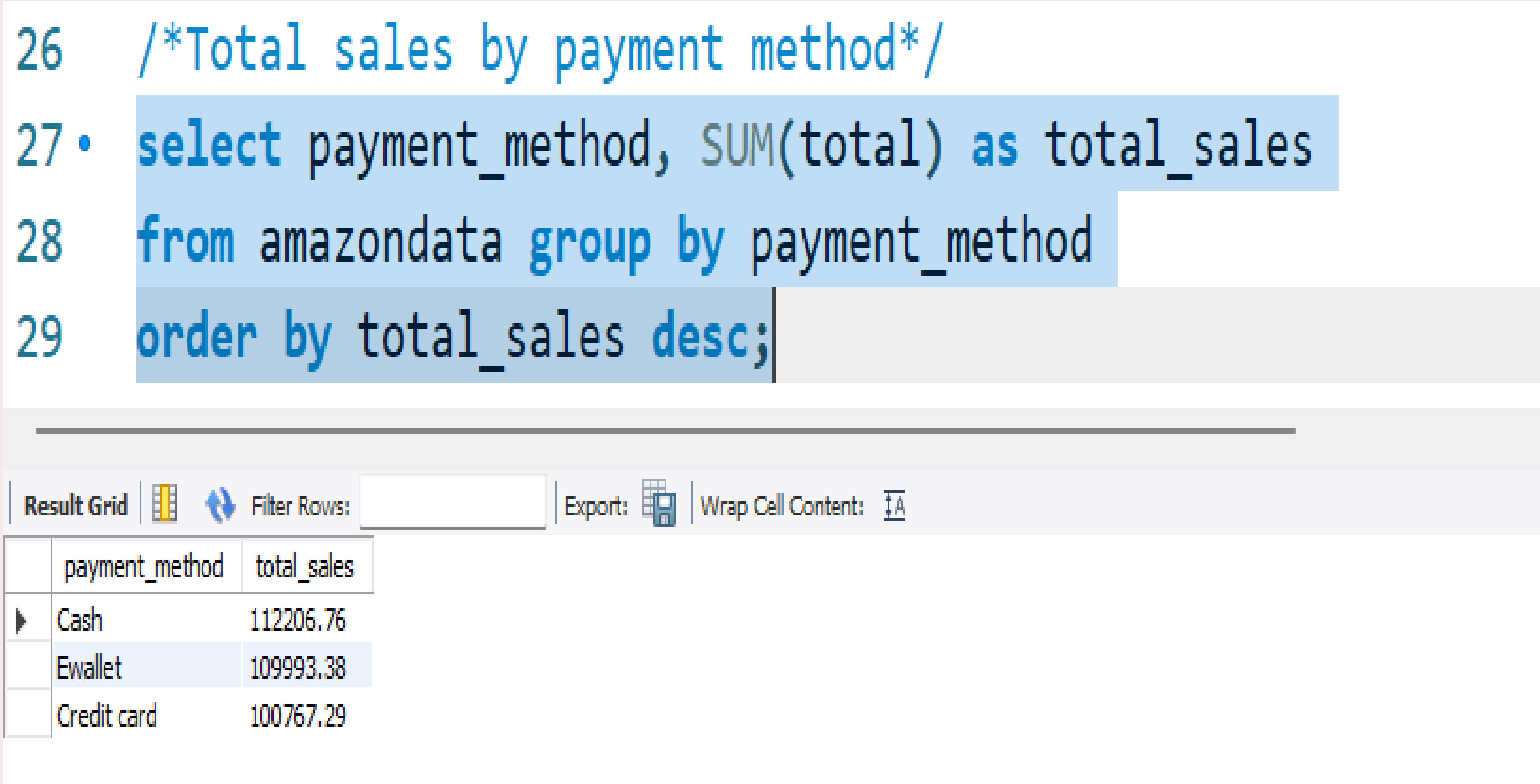
```
/* Total sales per month */
select MONTH(date) as month, SUM(total) as total_sales
from amazondata group by month
order by  month;
```

```
month  total_sales
1      116292.11
2      97219.58
3      109455.74
```



```
/* Total sales per branch */
select branch, SUM(total) as total_sales
from amazondata group by branch
order by total_sales desc;
```

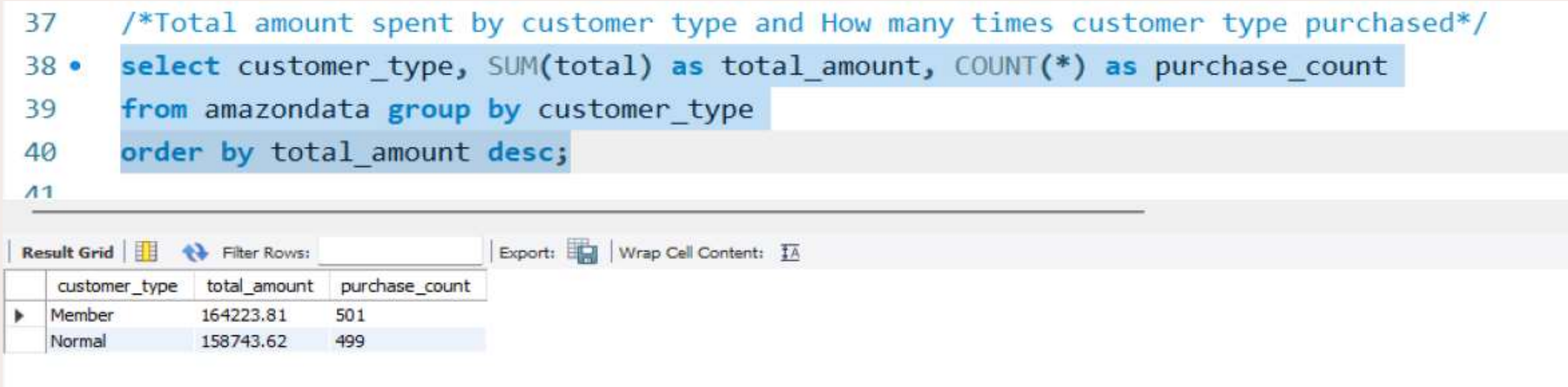
```
branch  total_sales
C       110568.86
A       106200.57
B       106198.00
```



```sql
/*Total sales by payment method*/
select payment_method, SUM(total) as total_sales
from amazondata group by payment_method
order by total_sales desc;
```

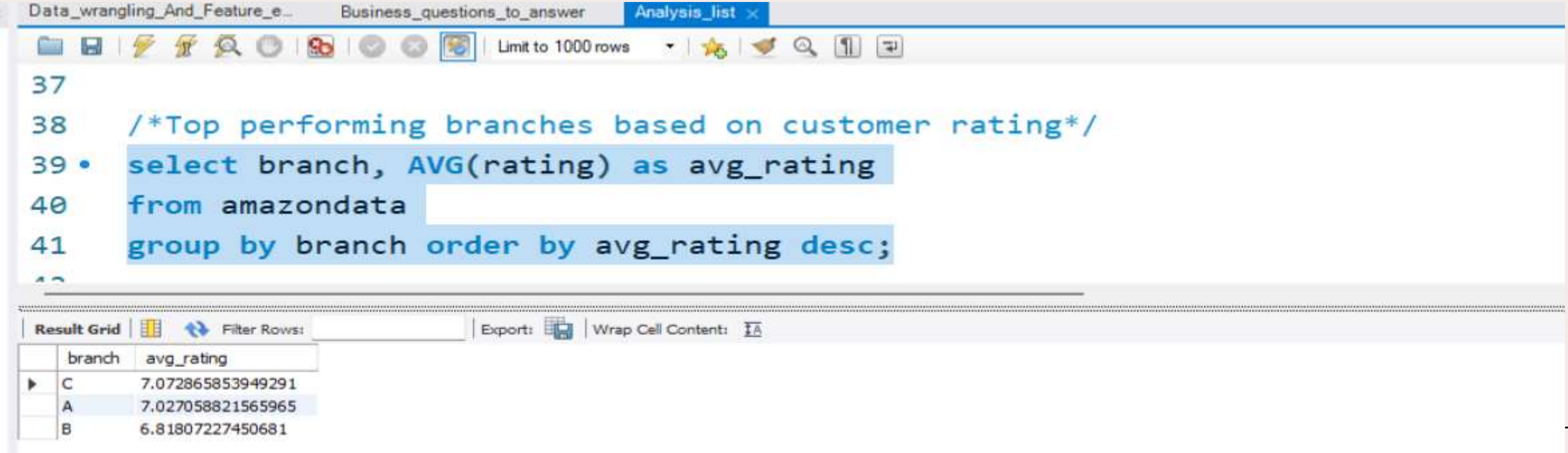
payment_method	total_sales
Cash	112206.76
Ewallet	109993.38
Credit card	100767.29

- Customer Analysis:



```
/*Total amount spent by customer type and How many times customer type purchased*/
select customer_type, SUM(total) as total_amount, COUNT(*) as purchase_count
from amazondata group by customer_type
order by total_amount desc;
```

customer_type	total_amount	purchase_count
Member	164223.81	501
Normal	158743.62	499



```
/*Top performing branches based on customer rating*/
select branch, AVG(rating) as avg_rating
from amazondata
group by branch order by avg_rating desc;
```

branch	avg_rating
C	7.072865853949291
A	7.027058821565965
B	6.81807227450681

Insights and Summary

Product analysis:

The top-performing product lines are dominated by the “Food and Beverages”, with total sales of 56,122.96, respectively On the other hand, the least performing product lines include Health and beauty, Home and lifestyle, and Fashion accessories, with total sales ranging from 49,193.84 to 56,144.96. These product lines may require attention and improvement strategies to boost sales.

Sales Analysis:

Monthly sales are relatively consistent, with the highest sales in January (116,292.11) and the lowest in February (97,219.58).- Branch C leads in terms of total sales (110,568.86), followed closely by Branch A and Branch B.- Cash payments account for the largest share of total sales (112,206.76), followed by E-wallet payments (109,993.38) and Credit Card payments (100,767.29).

Customer Analysis:

Members are the top-spending customer type, with a total spend of 164,223.81 and 501 purchases, closely followed by Normal customers with a total spend of 158,743.62 and 499 purchases.- Branch C is the top-performing branch based on customer ratings, with an average rating of 7.07, followed by Branch A (7.03) and Branch B (6.82).

Nikhil Karaka