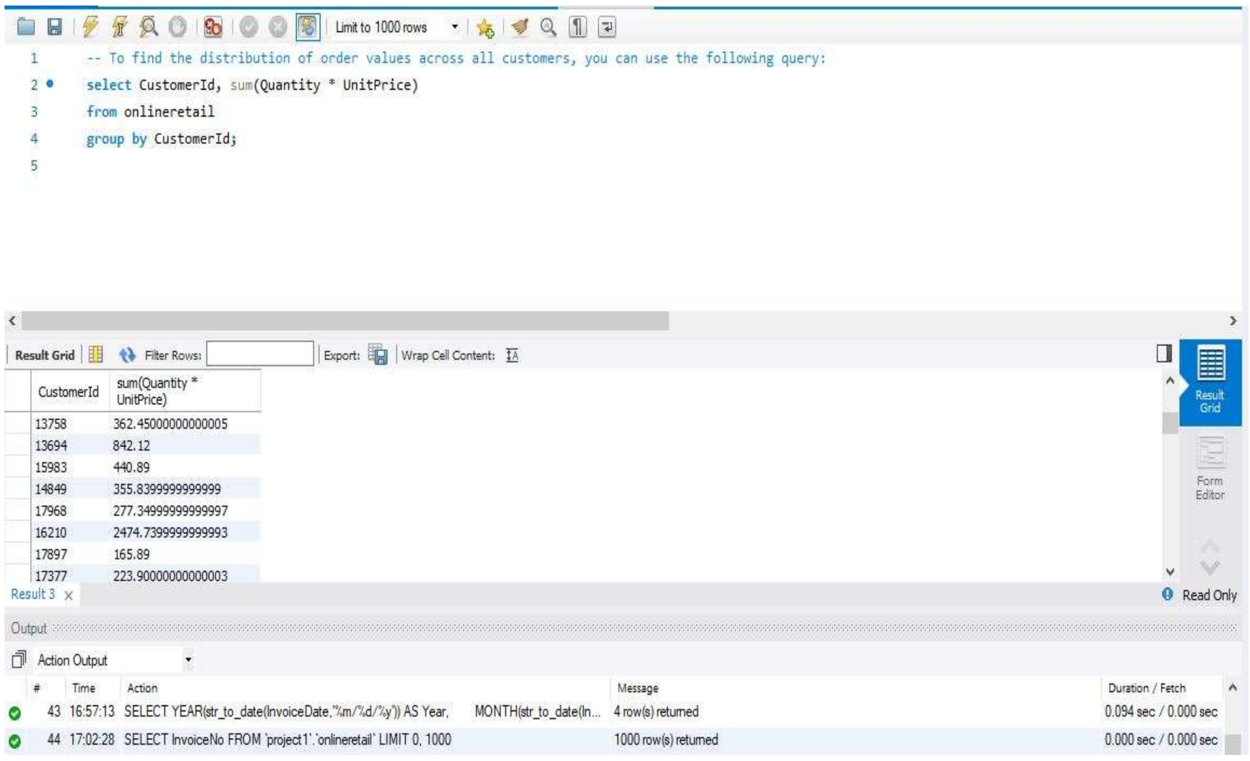
Distribution of Order Values:

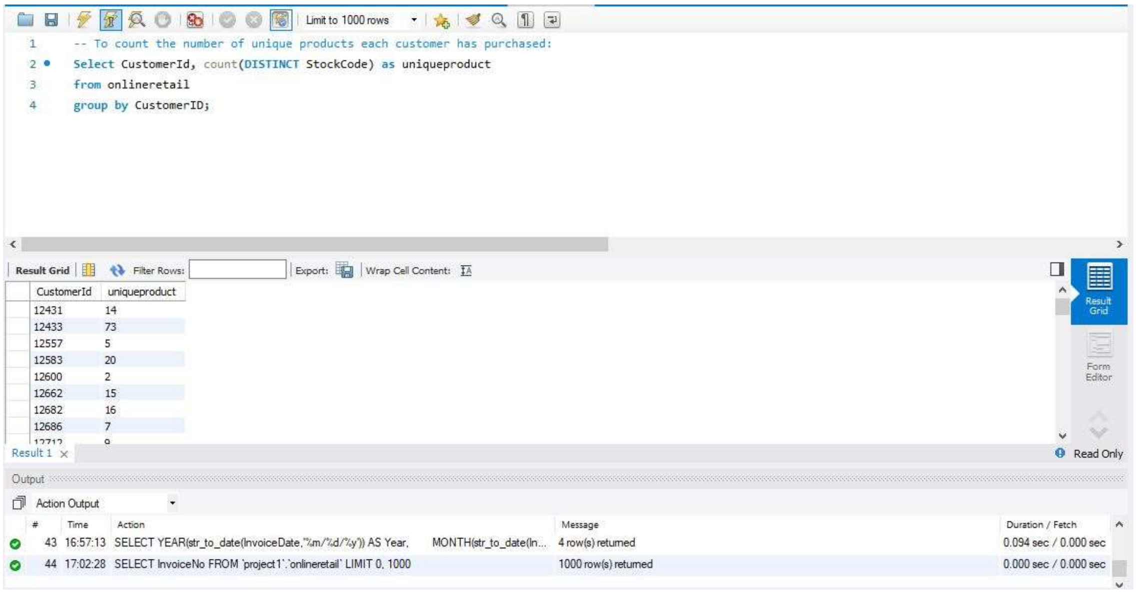


```
-- To find the distribution of order values across all customers, you can use the following query:
select CustomerId, sum(Quantity * UnitPrice)
from onlineretail
group by CustomerId;
```

CustomerId	sum(Quantity * UnitPrice)
13758	362.45000000000005
13694	842.12
15983	440.89
14849	355.8399999999999
17968	277.34999999999997
16210	2474.7399999999993
17897	165.89
17377	223.90000000000003

```
#   Time      Action                                                                          Message                 Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year,   MONTH(str_to_date(In...  4 row(s) returned       0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                     1000 row(s) returned    0.000 sec / 0.000 sec
```

Count Unique Product of Each Customer:

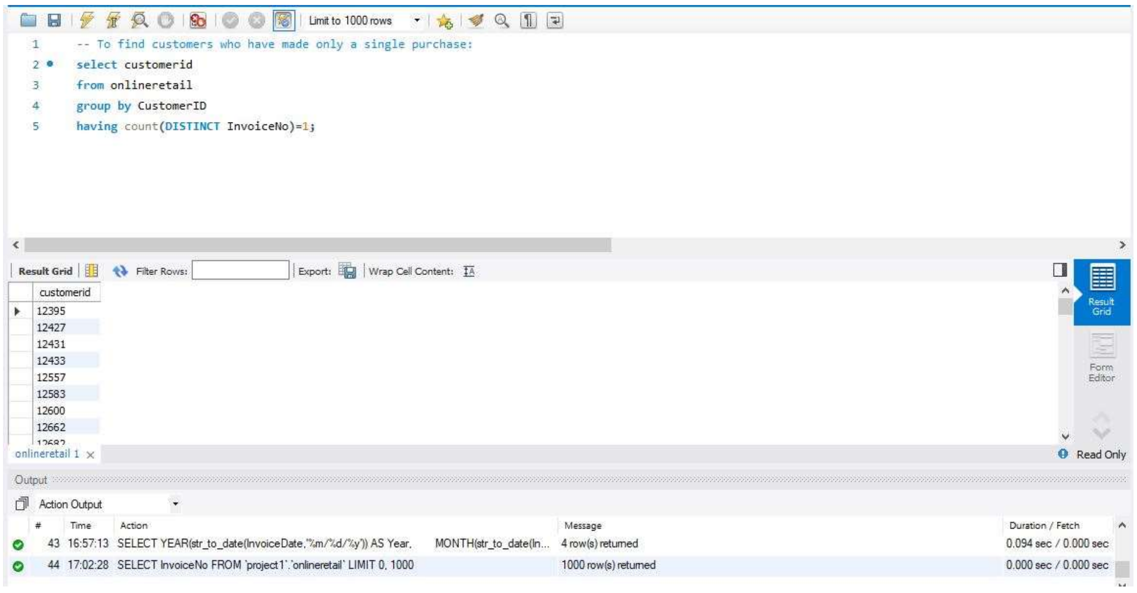


```sql
-- To count the number of unique products each customer has purchased:
Select CustomerId, count(DISTINCT StockCode) as uniqueproduct
from onlineretail
group by CustomerID;
```

CustomerId	uniqueproduct
12431	14
12433	73
12557	5
12583	20
12600	2
12662	15
12682	16
12686	7

```
#   Time      Action                                                                      Message               Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year,   MONTH(str_to_date(In...  4 row(s) returned     0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                        1000 row(s) returned  0.000 sec / 0.000 sec
```

Customer with Single Purchase:

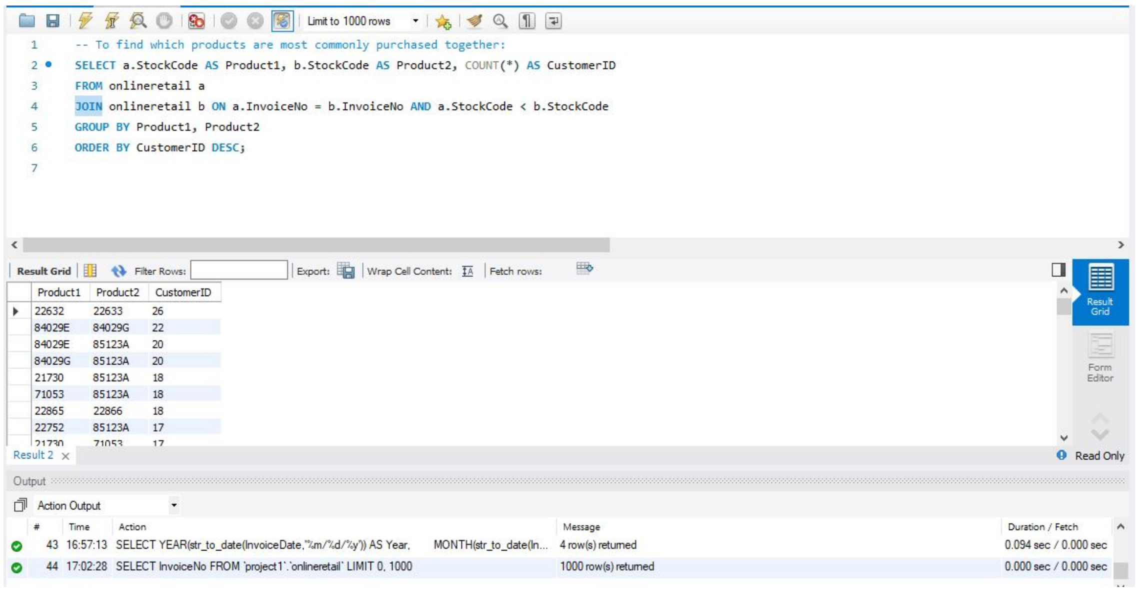


```sql
-- To find customers who have made only a single purchase:
select customerid
from onlineretail
group by CustomerID
having count(DISTINCT InvoiceNo)=1;
```

customerid
12395
12427
12431
12433
12557
12583
12600
12662

#	Time	Action	Message	Duration / Fetch
43	16:57:13	SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year, MONTH(str_to_date(In...	4 row(s) returned	0.094 sec / 0.000 sec
44	17:02:28	SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec

<u>**Commonly Purchased Products**</u>:

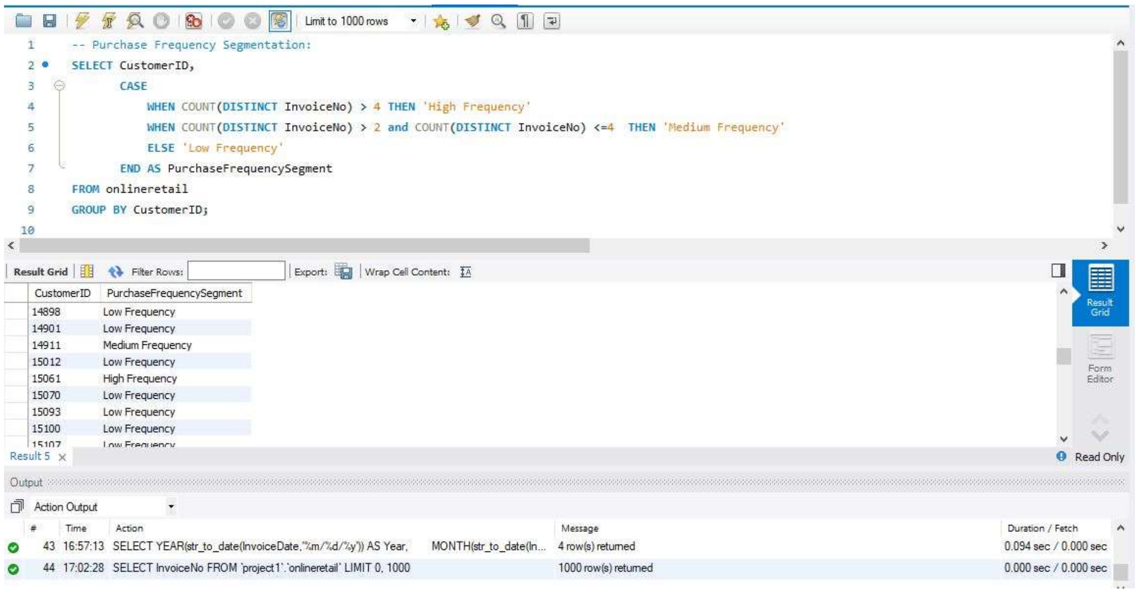


```sql
-- To find which products are most commonly purchased together:
SELECT a.StockCode AS Product1, b.StockCode AS Product2, COUNT(*) AS CustomerID
FROM onlineretail a
JOIN onlineretail b ON a.InvoiceNo = b.InvoiceNo AND a.StockCode < b.StockCode
GROUP BY Product1, Product2
ORDER BY CustomerID DESC;
```

Product1	Product2	CustomerID
22632	22633	26
84029E	84029G	22
84029E	85123A	20
84029G	85123A	20
21730	85123A	18
71053	85123A	18
22865	22866	18
22752	85123A	17
21730	71053	17

```
#   Time      Action                                                                          Message               Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year,   MONTH(str_to_date(In...  4 row(s) returned     0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                        1000 row(s) returned  0.000 sec / 0.000 sec
```

Purchased Frequency Segmentation:



```sql
-- Purchase Frequency Segmentation:
SELECT CustomerID,
       CASE
           WHEN COUNT(DISTINCT InvoiceNo) > 4 THEN 'High Frequency'
           WHEN COUNT(DISTINCT InvoiceNo) > 2 and COUNT(DISTINCT InvoiceNo) <=4  THEN 'Medium Frequency'
           ELSE 'Low Frequency'
       END AS PurchaseFrequencySegment
FROM onlineretail
GROUP BY CustomerID;
```

CustomerID	PurchaseFrequencySegment
14898	Low Frequency
14901	Low Frequency
14911	Medium Frequency
15012	Low Frequency
15061	High Frequency
15070	Low Frequency
15093	Low Frequency
15100	Low Frequency
15107	Low Frequency

Result 5

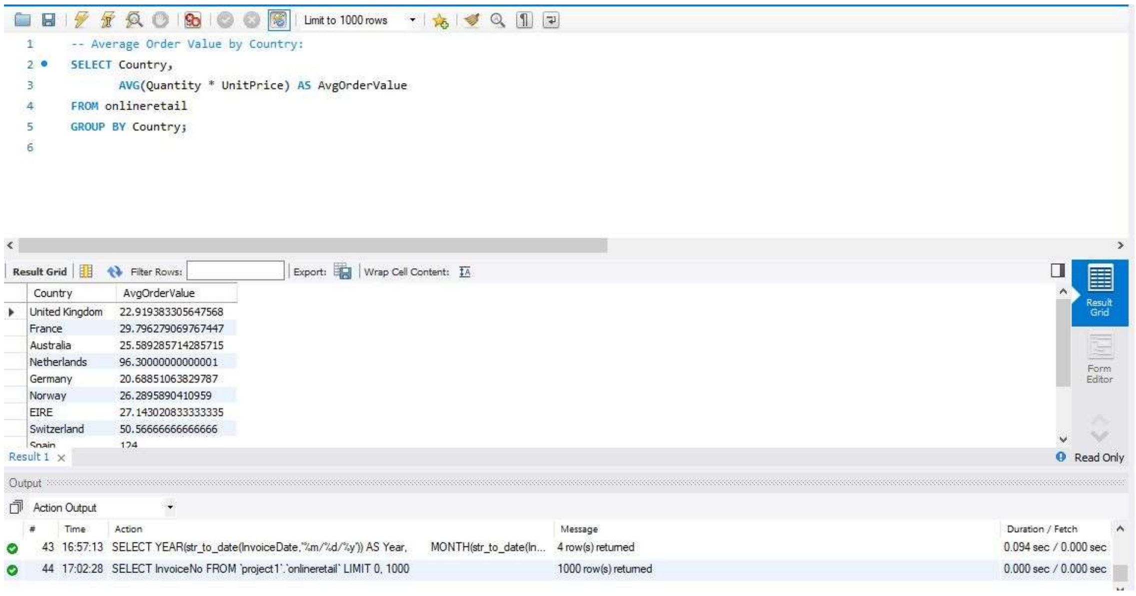
Read Only

Output

Action Output

```
#   Time      Action                                                                   Message               Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,"%m/%d/%y")) AS Year,   MONTH(str_to_date(In...  4 row(s) returned     0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000             1000 row(s) returned  0.000 sec / 0.000 sec
```

<u>**Average Order Value By Country:**</u>

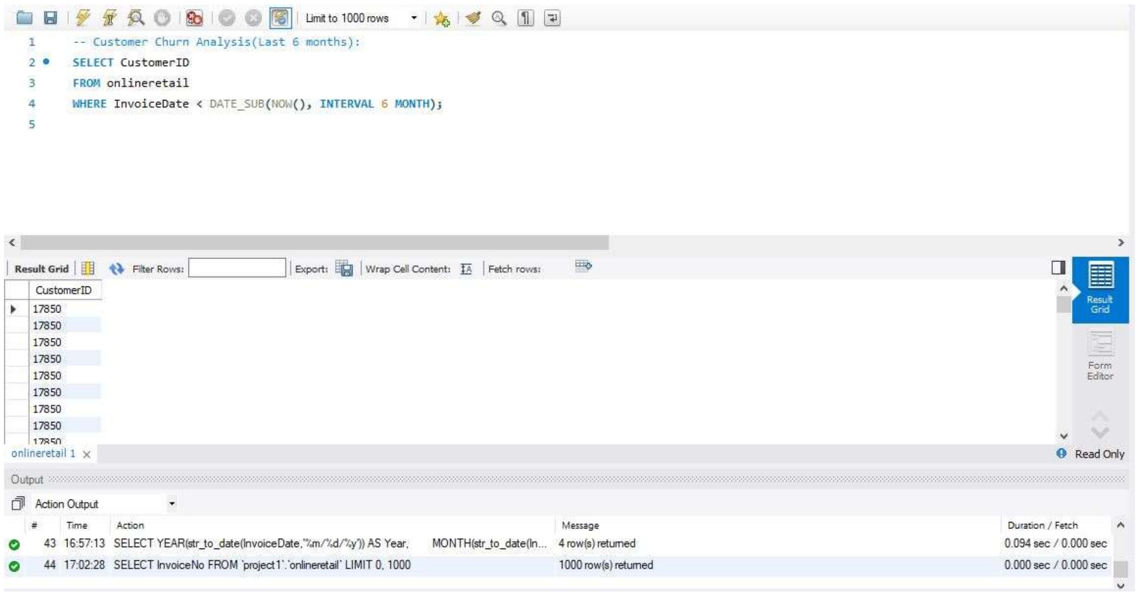


```sql
-- Average Order Value by Country:
SELECT Country,
       AVG(Quantity * UnitPrice) AS AvgOrderValue
FROM onlineretail
GROUP BY Country;
```

Country	AvgOrderValue
United Kingdom	22.919383305647568
France	29.796279069767447
Australia	25.589285714285715
Netherlands	96.30000000000001
Germany	20.68851063829787
Norway	26.2895890410959
EIRE	27.143020833333335
Switzerland	50.56666666666666
Spain	124

```
#   Time      Action                                                                            Message                Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,"%m/%d/%y')) AS Year,     MONTH(str_to_date(In...  4 row(s) returned      0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                          1000 row(s) returned   0.000 sec / 0.000 sec
```

Churn Analysis:

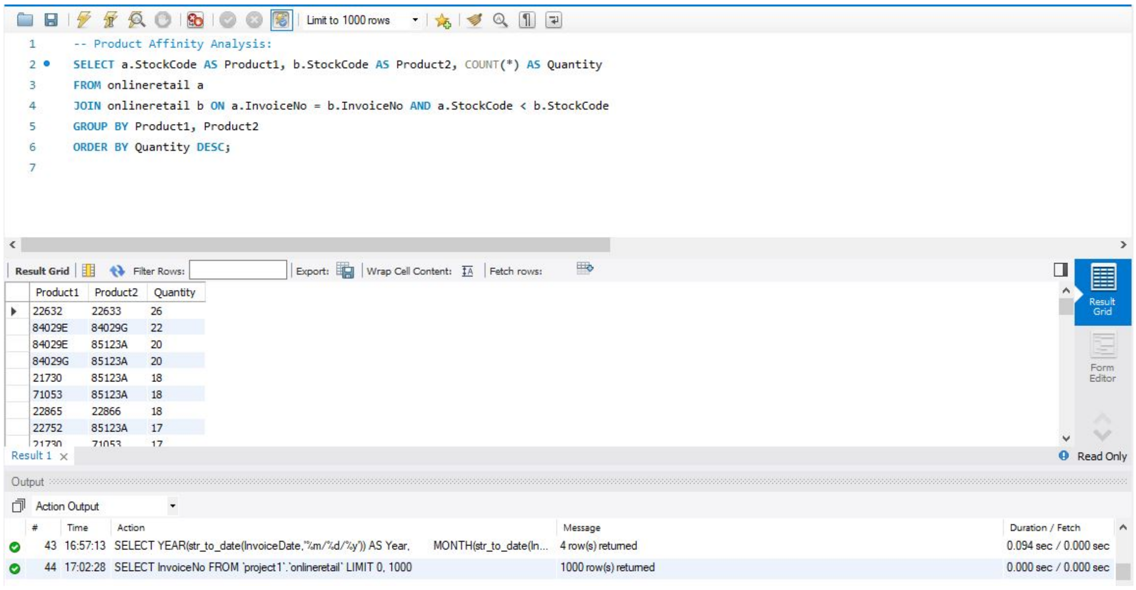


```sql
-- Customer Churn Analysis(Last 6 months):
SELECT CustomerID
FROM onlineretail
WHERE InvoiceDate < DATE_SUB(NOW(), INTERVAL 6 MONTH);
```

CustomerID
17850
17850
17850
17850
17850
17850
17850
17850
17850

```
#   Time      Action                                                                        Message              Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,"%m/%d/%y")) AS Year,   MONTH(str_to_date(In...   4 row(s) returned     0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                      1000 row(s) returned  0.000 sec / 0.000 sec
```

<u>Product Affinity Analysis:</u>

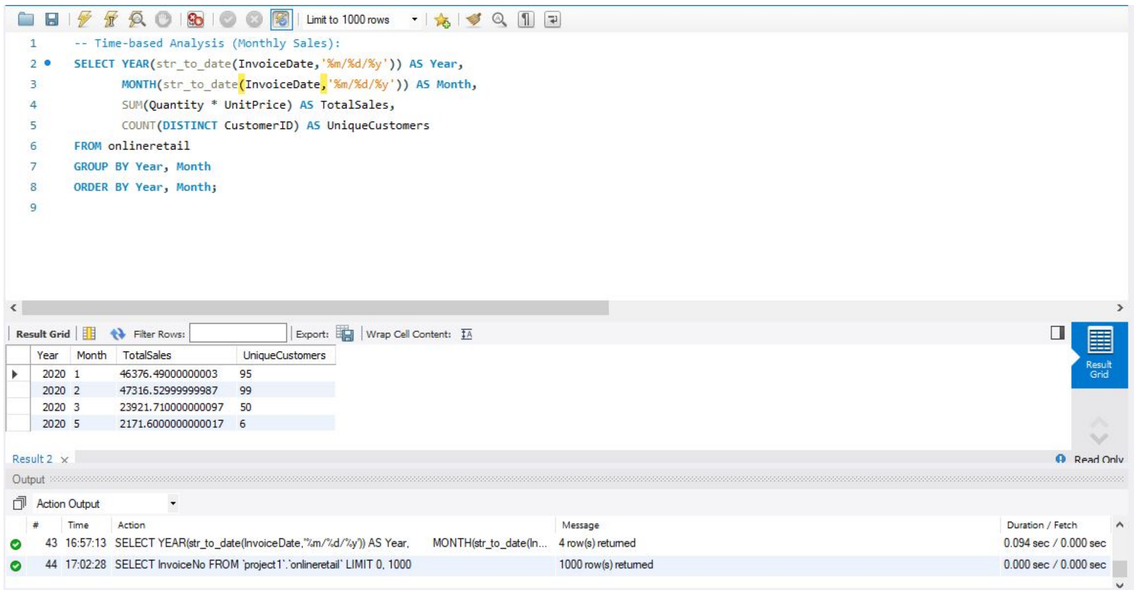


```
-- Product Affinity Analysis:
SELECT a.StockCode AS Product1, b.StockCode AS Product2, COUNT(*) AS Quantity
FROM onlineretail a
JOIN onlineretail b ON a.InvoiceNo = b.InvoiceNo AND a.StockCode < b.StockCode
GROUP BY Product1, Product2
ORDER BY Quantity DESC;
```

Product1	Product2	Quantity
22632	22633	26
84029E	84029G	22
84029E	85123A	20
84029G	85123A	20
21730	85123A	18
71053	85123A	18
22865	22866	18
22752	85123A	17
21730	71053	17

#	Time	Action	Message	Duration / Fetch
43	16:57:13	SELECT YEAR(str_to_date(InvoiceDate,"%m/%d/%y')) AS Year, MONTH(str_to_date(In...	4 row(s) returned	0.094 sec / 0.000 sec
44	17:02:28	SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec

Timed Based Analysis(Monthly Sales):



```sql
-- Time-based Analysis (Monthly Sales):
SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year,
       MONTH(str_to_date(InvoiceDate,'%m/%d/%y')) AS Month,
       SUM(Quantity * UnitPrice) AS TotalSales,
       COUNT(DISTINCT CustomerID) AS UniqueCustomers
FROM onlineretail
GROUP BY Year, Month
ORDER BY Year, Month;
```

Year	Month	TotalSales	UniqueCustomers
2020	1	46376.49000000003	95
2020	2	47316.52999999987	99
2020	3	23921.710000000097	50
2020	5	2171.6000000000017	6

```
#   Time      Action                                                                     Message             Duration / Fetch
43  16:57:13  SELECT YEAR(str_to_date(InvoiceDate,'%m/%d/%y')) AS Year,    MONTH(str_to_date(In...   4 row(s) returned   0.094 sec / 0.000 sec
44  17:02:28  SELECT InvoiceNo FROM `project1`.`onlineretail` LIMIT 0, 1000                    1000 row(s) returned  0.000 sec / 0.000 sec
```

THE END