

# SQL Queries

## Table1

- 1) Write a query to **Retrieve the Top 20 Most Expensive Products.**

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select * from table1 order by Price desc limit 20;
```

The result grid displays the top 20 most expensive products:

SI_No	Brand_Name	Category	Product_Type	Product_Name	Product_Code	Price
3424	Addis	Unisex	Top	REAL MADRID TIRO 23 ALL-WEATHER JACKET	HB15110	49999
3423	Addis	Unisex	Top	REAL MADRID TIRO 23 ALL-WEATHER JACKET	HB15090	49999
3416	Addis	Unisex	Top	REAL MADRID TIRO 23 ALL-WEATHER JACKET	HB14950	49999
3432	Addis	Unisex	Top	REAL MADRID TIRO 23 ALL-WEATHER JACKET	HB15270	49999
3422	Addis	Unisex	Top	REAL MADRID TIRO 23 ALL-WEATHER JACKET	HB15070	49999

The output pane shows the execution log with the following messages:

```
80 21:14:08 PREPARE stmt FROM INSERT INTO 'capstone_project_data'.table1 ('SI_No', Br... OK 0.000 sec
81 21:14:42 DEALLOCATE PREPARE stmt OK 0.000 sec
82 21:15:01 use capstone_project_data 0 row(s) affected 0.000 sec
83 21:15:01 select * from table1 order by Price desc LIMIT 0, 1000 1000 row(s) returned 0.032 sec / 0.000 sec
84 21:15:27 use capstone_project_data 0 row(s) affected 0.000 sec
85 21:15:27 select * from table1 order by Price desc limit 20 20 row(s) returned 0.032 sec / 0.000 sec
```

- 2) Write a query to **Calculate the Total Price for Each Category.**

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select Category, sum(Price) as Total_Price from table1 group by Category;
```

The result grid displays the total price for each category:

Category	Total_Price
Kids	3467158
Women	3110467
Men	8552072
Unisex	3338395

The output pane shows the execution log with the following messages:

```
84 21:15:27 use capstone_project_data 0 row(s) affected 0.000 sec
85 21:15:27 select * from table1 order by Price desc limit 20 20 row(s) returned 0.032 sec / 0.000 sec
86 21:15:47 use capstone_project_data 0 row(s) affected 0.000 sec
87 21:15:47 select Category, sum(Price) as Total_Price from table1 group by Category LIMIT 0, ... Error Code: 1630. FUNCTION capstone_project_data.sum does not exist. Check th... 0.000 sec
88 21:15:11 use capstone_project_data 0 row(s) affected 0.000 sec
89 21:15:11 select Category, sum(Price) as Total_Price from table1 group by Category LIMIT 0, ... 4 row(s) returned 0.032 sec / 0.000 sec
```

### 3) Write a query to Retrieve Products with a Price Range 2000 to 5000

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select * from table1 where Price between 2000 and 5000;
```

The Result Grid displays the following data:

SI_No	Brand_Name	Category	Product_Type	Product_Name	Product_Code	Price
3976	Stop	Men	Half Sleeves	Solid Cotton Regular Mens T-Shirt	S23TCMANDPLWHC	2320
3977	Zink London	Women	Cotton	Solid Cotton Regular Women's Top	S23ZINKT05117	3457
3979	Intune	Men	Trousers	Solid Cotton Relaxed Fit Men's Casual Pant	A23M-TRCRON135	4563
3980	Intune	Men	Solid	Solid Cotton Relaxed Fit Men's Casual Pant	A23M-TRCRON137	4563
3981	Dolce Crudo	Women	Cotton	Solid Cotton Relaxed Fit Womens Shorts	D052ZDEN909115	3456

The Output pane shows the execution log with the following messages:

- 86 21:34:47 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 87 21:34:47 select Category, sum (Price) as Total\_Price from table1 group by Category LIMIT 0, ... Error Code: 1630. FUNCTION capstone\_project\_data sum does not exist. Check th... 0.000 sec
- 88 21:35:11 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 89 21:35:11 select Category, sum(Price) as Total\_Price from table1 group by Category LIMIT 0, ... 4 row(s) returned 0.032 sec / 0.000 sec
- 90 21:36:34 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 91 21:36:34 select \* from table1 where Price between 2000 and 5000 LIMIT 0, 1000 1000 row(s) returned 0.000 sec / 0.016 sec

### 4) Write a query to Calculate the Average Price for Each Brand

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select Brand_Name, avg(Price) as Average_Price from table1 group by Brand_Name;
```

The Result Grid displays the following data:

Brand_Name	Average_Price
Stop	3782.1803
Intune	5418.5833
Life	5382.6216
Global Desi Girls	4756.8000
Amante	3456.0000

The Output pane shows the execution log with the following messages:

- 88 21:35:11 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 89 21:35:11 select Category, sum(Price) as Total\_Price from table1 group by Category LIMIT 0, ... 4 row(s) returned 0.032 sec / 0.000 sec
- 90 21:36:34 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 91 21:36:34 select \* from table1 where Price between 2000 and 5000 LIMIT 0, 1000 1000 row(s) returned 0.000 sec / 0.016 sec
- 92 21:39:47 use capstone\_project\_data 0 row(s) affected 0.000 sec
- 93 21:39:47 select Brand\_Name, avg(Price) as Average\_Price from table1 group by Brand\_Nam... 170 row(s) returned 0.047 sec / 0.000 sec

## 5) Write a query to find Products with Unique Brand-Category Combinations

The screenshot shows MySQL Workbench with a query executed on the 'capstone\_project\_data' database. The query is as follows:

```
1 use capstone_project_data;
2 select Brand_Name,Category, count(Product_Name) as Product_count
3 from table1 group by Brand_Name,Category;
```

The result grid shows the following data:

Brand_Name	Category	Product_count
Stop	Kids	319
Intune	Kids	18
Life	Women	19
Life	Kids	143
Global Desi Girls	Kids	5

The output pane shows the execution details of the query, including the time taken and the number of rows returned.

## Table 2

### 1) Write a query Order Rows by Material Type in Ascending Order .

The screenshot shows MySQL Workbench with a query executed on the 'capstone\_project\_data' database. The query is as follows:

```
1 use capstone_project_data;
2 select * from table2 order by Material_Type;
```

The result grid shows the following data:

SI_No	Color	Wear_Type	Material_Type	Reviews
4625	Green	Bottom Wear	Cotton	10
4577	Blue	Bottom Wear	Cotton	10
4869	Blue	Top Wear	Cotton	10
4868	Green	Top Wear	Cotton	10
4867	White	Top Wear	Cotton	10

The output pane shows the execution details of the query, including the time taken and the number of rows returned.

## 2) Write a query to Calculate the Average Number of Reviews

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select Avg(Reviews) from table2;
```

The result grid shows the output of the query:

Avg(Reviews)
9.2586

The bottom panel shows the output log with the following entries:

#	Time	Action	Message	Duration / Fetch
107	21:53:32	PREPARE stmt FROM INSERT INTO 'capstone_project_data'.table2 ('SI_No', 'C...	OK	0.000 sec
108	21:54:01	DEALLOCATE PREPARE stmt	OK	0.000 sec
109	21:56:25	use capstone_project_data	0 row(s) affected	0.000 sec
110	21:56:25	select * from table2 order by Material_Type LIMIT 0, 1000	1000 row(s) returned	0.032 sec / 0.000 sec
111	21:59:12	use capstone_project_data	0 row(s) affected	0.000 sec
112	21:59:12	select Avg(Reviews) from table2 LIMIT 0, 1000	1 row(s) returned	0.016 sec / 0.000 sec

## 3) Write a query to Retrieve the Top 30 Most Expensive Products by Weartype:

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color, a.wear_Type, a.Material_type, a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 SELECT *
8 FROM ( SELECT *, ROW_NUMBER() OVER (PARTITION BY wear_type ORDER BY price DESC) as row_num
9 FROM abc
10 ) as ranked
11 WHERE row_num <= 30
12 ORDER BY wear_type, price DESC;
```

The result grid shows the output of the query:

SI_No	Brand_Name	Category	Product_Type	Product_Name	Product_Code	Price	color	wear_Type
3985	Infuse	Kids	Slim Fit	Solid Cotton Slim Fit Boys Jeans	INFBJNRS100A	8990	BLACK	Bottom Wea
3912	Stop	Kids	Solid	Solid Cotton Regular Fit Boy's Shorts	S237100SN6HRT04	8990	NAVY	Bottom Wea
3890	Life	Kids	Solid	Solid Cotton Regular Fit Boys Joggers	S227118WJ004	8990	RUST	Bottom Wea
3909	Stop	Kids	Solid	Solid Cotton Regular Fit Boy's Shorts	S237108WSHRT01	8990	KHAKI	Bottom Wea



#### 4) Write a query to Find Products with Similar Colors.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 SELECT Color,Product_Name from abc order by color;
```

The result grid displays the following data:

color	Product_Name
AQUA	Printed Polyester Mens Festive Wear Kurta
AQUA	Solid Blended Regular Fit Boys Kurta Pyjama Set
AQUA	Printed Georgette Round Neck Girls Gown
AQUA	Solid Chandori Regular Fit Girls Kurta Dupatta Set
AQUA	Solid Cotton Men's Festive Wear Kurta
Aqua	FAME FOREVER Girls Textured Elasticated Joggers

The Action Output pane shows the following messages:

#	Time	Action	Message	Duration / Fetch
130	22:15:50	WITH abc AS ( SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews	50 row(s) returned	0.109 sec / 0.000 sec
131	22:17:48	use capstone_project_data	0 row(s) affected	0.000 sec

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 SELECT Color,count(Product_Name) as Products from abc group by color;
```

The result grid displays the following data:

color	Products
MINT	18
WHITE	396
MID	27
PEACH	52
PINK	364
PURPLE	117

The Action Output pane shows the following messages:

#	Time	Action	Message	Duration / Fetch
140	22:27:15	use capstone_project_data	0 row(s) affected	0.000 sec
141	22:27:15	with abc as ( SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews	105 row(s) returned	0.046 sec / 0.000 sec

5) Write a query to **Calculate the Total Price for Each Material Type**.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 SELECT Material_Type,sum(Price) as Total_Price from abc group by Material_Type;
8
9
10

```

The Result Grid shows the following data:

Material_type	Total_Price
Cotton	2091317
Denim	571745
Polyester	11213024
Woven	878421
Cotton	1178454
Polyester	304390

The Action Output shows the execution of the query, indicating that 105 row(s) were returned.

## Table 3

1) Write a query to **Order Rows by Return Time in Ascending Order**.

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```

1 use capstone_project_data;
2 select * from table3 order by ReturnTime;

```

The Result Grid shows the following data:

Sl_No	Size_1	Size_2	Size_3	Size_4	Country_Of_Origin	ReturnTime	Occasion
4041	2-3 Y	3-4 Y	5-6 Y	7-8 Y	India	0	Casual
4042	11-12 Y	13-14 Y	15-16 Y	9-10 Y	India	0	Casual
4046					India	0	Casual
4047					India	0	Casual
4048	11-12 Y	13-14 Y	15-16 Y	9-10 Y	India	0	Casual

The Action Output shows the execution of the query, indicating that 1000 row(s) were returned.

2) Write a query to find distinct country of origin

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select distinct Country_Of_Origin from table3;
```

The left sidebar shows the 'SCHEMAS' panel with a tree view of the database structure. The 'table3' table is selected, and its columns are listed: SI\_No, Size\_1, Size\_2, Size\_3, Size\_4, Country\_Of\_Origin, ReturnTime, and Occasion. The 'Country\_Of\_Origin' column is highlighted.

The 'Result Grid' shows the output of the query:

Country_Of_Origin
India
America
China
Bangladesh

The bottom status bar shows the query execution details: 153 22:40:53 use capstone\_project\_data (0 row(s) affected, 0.000 sec) and 154 22:40:53 select distinct Country\_Of\_Origin from table3 LIMIT 0, 1000 (4 row(s) returned, 0.031 sec / 0.000 sec).

3) Write a query to **Calculate the Average Return Time by Country of Origin**

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select Country_Of_Origin, avg(ReturnTime) from table3 group by Country_Of_Origin;
```

The left sidebar shows the 'SCHEMAS' panel with a tree view of the database structure. The 'table3' table is selected, and its columns are listed: SI\_No, Size\_1, Size\_2, Size\_3, Size\_4, Country\_Of\_Origin, ReturnTime, and Occasion. The 'Country\_Of\_Origin' column is highlighted.

The 'Result Grid' shows the output of the query:

Country_Of_Origin	avg(ReturnTime)
India	8.8695
America	7.0000
China	14.0000
Bangladesh	14.0000

The bottom status bar shows the query execution details: 165 22:48:40 use capstone\_project\_data (0 row(s) affected, 0.000 sec) and 166 22:48:40 select Country\_Of\_Origin, avg(ReturnTime) from table3 group by Country\_Of\_Origin (4 row(s) returned, 0.031 sec / 0.000 sec).

4) Write a query to **Find Occasions with High Return Times.**

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select Occasion, ReturnTime from table3 order by ReturnTime desc;
```

The result grid displays the following data:

Occasion	ReturnTime
Casual	14
Casual	14
Casual	14
Casual	14
Casual	14

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
167	22:51:06	use capstone_project_data	0 row(s) affected	0.000 sec
168	22:51:06	select Occasion, ReturnTime from table3 order by ReturnTime desc LIMIT 0, 1000	1000 row(s) returned	0.031 sec / 0.000 sec

5) Write a query to **Retrieve Products with Multiple Sizes (Size1, Size2, Size3, Size4):**

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select * from table3 where Size_1 not in(' ','-') and Size_2 not in(' ','-')
3 and size_3 not in(' ','-') and size_4 not in(' ','-');
```

The result grid displays the following data:

SI_No	Size_1	Size_2	Size_3	Size_4	Country_Of_Origin	ReturnTime	Occasion
2	2-3 Y	3-4 Y	5-6 Y	7-8 Y	India	14	Casual
3	30	28	28	32	India	0	Casual
20	5-8 YEARS	7-9 YEARS	11-12 YEARS	12-14 YEARS	India	14	Casual
37	8-9 Y	9-10 Y	11-12 Y	13-14 Y	India	14	Casual
38	8-9 Y	9-10 Y	11-12 Y	13-14 Y	India	14	Casual

The output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
173	22:58:10	use capstone_project_data	0 row(s) affected	0.000 sec
174	22:58:10	select * from table3 where Size_1 not in(' ','-') and Size_2 not in(' ','-') and size_3 not in(' ','-') and size_4 not in(' ','-')	1000 row(s) returned	0.016 sec / 0.000 sec



# JOIN QUERY using table1,table2 and table3

- 1) Write a query find Brandname, Category , Producttype ,ProductName with a **listed price greater than the average listed price** in table 1.

The screenshot shows MySQL Workbench with a query executed in the SQL Editor. The query is:

```
1. use capstone_project_data;
2. select Brand_Name,Category,Product_Type,Product_Name from table1 where Price > (select avg(Price) from table1);
```

The result grid shows the following data:

Brand_Name	Category	Product_Type	Product_Name
Intune	Kids	Cotton	A Day at the Beach Cotton T-shirt for Boys
Life	Kids	Half Sleeves	A20711GTEE204_PEACH
Stop	Kids	Tops	A227100SNGTEE03_PINK
Stop	Kids	Tops	A227100SNGTEE04_PURPLE
Stop	Kids	Printed	A22710GTEE07_OFF_WHITE

The bottom status bar indicates that 1000 row(s) were returned.

- 2) Write a query to list all Brandname, Category , Producttype ,ProductName along with their corresponding **color information** from Table2

The screenshot shows MySQL Workbench with a query executed in the SQL Editor. The query is:

```
1. use capstone_project_data;
2. select a.Brand_Name,a.Category,a.Product_Type,a.Product_Name,b.color
3. from table1 a join table2 b on a.SI_No=b.SI_No
```

The result grid shows the following data:

Brand_Name	Category	Product_Type	Product_Name	color
Stop	Kids	Tops	206955264_MINT	MINT
Intune	Kids	Cotton	A Day at the Beach Cotton T-shirt for Boys	WHITE
Life	Women	Cotton Stretch	A Line Above Knee Denim Women's Casual Wea...	MID
Life	Kids	Half Sleeves	A20711GTEE204_PEACH	PEACH
Stop	Kids	Tops	A227100SNGTEE03_PINK	PINK

The bottom status bar indicates that 1000 row(s) were returned.

3) Write a query to find the **average reviews** for each Producttype ,ProductName using table 1 and table 2

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 select Product_Name,Product_Type,avg(Reviews) from abc
8 group by Product_Name,Product_Type;
```

The Result Grid shows the following data:

Product_Name	Product_Type	avg(Reviews)
U.S. POLO ASSN. Men Solid Regular Fit Casual ...	Trousers	10.0000
U.S. POLO ASSN. Men Solid Regular Fit Casual ...	Flat Front	10.0000
U.S. POLO ASSN. Men Solid Regular Fit Polo T-s...	Polo Neck	10.0000
U.S. POLO ASSN. Men Solid Regular Fit Polo T-s...	Polo T-shirt	10.0000
U.S. POLO ASSN. Men Solid Regular Fit Polo T-s...	T-Shirt	10.0000

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	09:59:47	with abc as ( SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews	Error Code: 1045. No database selected Select the default DB to be used by doubl...	0.000 sec
2	10:00:01	use capstone_project_data	0 row(s) affected	0.000 sec
3	10:00:01	with abc as ( SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews	3162 row(s) returned	0.078 sec / 0.000 sec

4) Write a query to find products with a Product name, **material type** that matches the **most common material type** in (use table1 and table 2)

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 with abc as (
3     SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews
4     FROM table2 a
5     LEFT JOIN table1 b ON a.sl_no = b.sl_no
6 )
7 select Product_Name,Material_Type from abc where Material_Type = ( select max(Material_Type) from abc);
```

The Result Grid shows the following data:

Product_Name	Material_Type
Abstract Viscose Relaxed Fit Men's Casual Shirt	Woven
Abstract Viscose Relaxed Fit Men's Casual Shirt	Woven
Abstract Viscose Relaxed Fit Men's Casual Shirt	Woven
Block Cotton Regular Mens T-Shirt	Woven
Block Cotton Regular Mens T-Shirt	Woven

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
191	23:19:48	use capstone_project_data	0 row(s) affected	0.000 sec
192	23:19:48	with abc as ( SELECT b.*, a.color,a.wear_Type,a.Material_type,a.Reviews	419 row(s) returned	0.047 sec / 0.016 sec

- 5) Write a query to list all products Weartype , Material type along with their corresponding **sizes** from (use table 2 and table 3)

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use capstone_project_data;
2 select a.Wear_Type,a.Material_Type,b.Size_1,b.Size_2,b.Size_3,b.Size_4
3 from table2 a join table3 b on a.SI_No=b.SI_No;
4
5
6
7
8
```

The result grid displays the following data:

Wear_Type	Material_Type	Size_1	Size_2	Size_3	Size_4
Top Wear	Cotton	7-8Y	-	2-3Y	3-4Y
Top Wear	Cotton	2-3 Y	3-4 Y	5-6 Y	7-8 Y
Top Wear	Denim	30	28	28	32
Top Wear	Cotton	9-10Y	-	11-12Y	13-14Y
Top Wear	Cotton	7-8 Y	-	2-3 Y	3-4 Y

The output pane shows the execution of the query, indicating that 1000 rows were returned.

- 6) Write a query to find the **average return time** for each product type (use table1 and table 3)

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
9 with abc as (
10     SELECT a.*, b.ReturnTime
11     FROM table1 a
12     LEFT JOIN table3 b ON a.sI_no = b.sI_no
13 )
14 select Product_Type, avg(ReturnTime) from abc group by Product_Type;
15
16
17
18
19
```

The result grid displays the following data:

Product_Type	avg(ReturnTime)
Tops	7.4000
Cotton	9.8824
Cotton Stretch	2.0000
Half Sleeves	9.4889
Printed	6.3636

The output pane shows the execution of the query, indicating that 326 rows were returned.

- 7) Write a query to find Country of origin, return time ,occasion of each product type .  
(use table 1 and table 3)

The screenshot shows the MySQL Workbench interface. The SQL Editor contains the following query:

```
1 use capstone_project_data;
2 select a.Country_Of_Origin,a.ReturnTime,a.Occasion,b.Product_Type
3 from table3 a join table1 b on a.SI_No=b.SI_No
4 group by a.Country_Of_Origin, a.ReturnTime, a.Occasion, b.Product_Type;
```

The Result Grid displays the following data:

Country_Of_Origin	ReturnTime	Occasion	Product_Type
India	14	Casual	Tops
India	14	Casual	Cotton
India	0	Casual	Cotton Stretch
India	0	Casual	Half Sleeves
India	0	Casual	Tops

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
199	23:38:00	use capstone_project_data	0 row(s) affected	0.000 sec
200	23:38:00	select a.Country_Of_Origin,a.ReturnTime,a.Occasion,b.Product_Type from table3 a...	470 row(s) returned	0.047 sec / 0.000 sec