

Checking the Customers Table:

The screenshot shows the MySQL Workbench interface. On the left, the Navigator pane displays the 'new_schema' with tables 'customers', 'ecommerce_shipping_data', 'olist_customers_dataset', 'products', and 'transactions'. The 'customers' table is selected. The main editor shows the SQL query: `select * from customers;`. The Result Grid displays the following data:

CustomerID	CustomerName	Region	SignupDate
C0191	Sanantha Gibson	South America	2024-04-07
C0192	Sarah Arias	South America	2022-09-15
C0193	Douglas Torres	Asia	2023-02-03
C0194	Stacy Cook	North America	2024-05-04
C0195	Jeremy McLaughlin	South America	2024-09-17
C0196	Laura Watts	Europe	2022-06-07
C0197	Christina Harvey	Europe	2023-03-21
C0198	Rebecca Ray	Europe	2022-02-27
C0199	Andrea Jenkins	Europe	2022-12-03
C0200	Kelly Cross	Asia	2023-06-11

The bottom pane shows the Action Output with the following messages:

- 41 12:01:06 SHOW SESSION VARIABLES LIKE 'lower_case_table_names' OK
- 42 12:01:06 SHOW TABLES FROM 'new_schema' like 'transactions' OK
- 43 12:01:11 CREATE TABLE 'new_schema'.'transactions' (TransactionID text, CustomerID text, ProductID text, TransactionDate text) OK
- 44 12:01:11 PREPARE stmt FROM 'INSERT INTO 'new_schema'.'transactions' (TransactionID, CustomerID, ProductID, TransactionDate) VALUES (1, 'C0191', 'C0191', '2024-04-07')' OK
- 45 12:15:53 DEALLOCATE PREPARE stmt OK
- 46 12:16:38 select * from customers LIMIT 0, 1000 200 row(s) returned

Customers from the 'Asia' region, sorted by signup date:

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

```
1 select CustomerName, Region, SignupDate
2 from customers
3 where Region='Asia'
4 Order by SignupDate DESC;
```

The Result Grid displays the following data:

CustomerName	Region	SignupDate
Julia Palmer	Asia	2024-11-13
Joshua Hamilton	Asia	2024-11-11
Cynthia Clayton	Asia	2024-11-01
Bruce Rhodes	Asia	2024-09-29
Elizabeth Wells	Asia	2024-09-21
Belinda Garner	Asia	2024-09-17
Brittany Harvey	Asia	2024-09-04
Nancy Walker	Asia	2024-07-13
Jennifer Pena	Asia	2024-06-29
Julia Kelly	Asia	2024-06-01
Jacqueline Zam...	Asia	2024-05-29

Count of customers by region:

The screenshot shows a database IDE with a SQL query editor and a result grid. The query is as follows:

```
1 SELECT region, COUNT(*) AS total_customers
2 FROM customers
3 GROUP BY region;
```

The result grid displays the following data:

region	total_customers
South America	59
Asia	45
North America	46
Europe	50

Customers with their purchases;

The screenshot shows a database IDE with a SQL query editor and a result grid. The query is as follows:

```
1 SELECT c.CustomerName, t.i>TransactionID, t.TotalValue
2 FROM customers c
3 INNER JOIN transactions t ON c.i>CustomerID = t.CustomerID
4 LIMIT 0, 1000;
```

The result grid displays the following data:

CustomerName	i>TransactionID	TotalValue
Andrea Jenkins	T00001	300.68
Brittany Harvey	T00112	300.68
Kathryn Stevens	T00166	300.68
Travis Campbell	T00272	601.36
Timothy Perez	T00363	902.04
Anna Ball	T00442	300.68
Jeremy McLaughlin	T00490	902.04
David Li	T00536	300.68
Miguel Wong	T00564	902.04
Robert Jones	T00631	601.36
Nicholas Ellis	T00727	902.04

All customers, even those who haven't purchased:

Checking Customers Table SQL File 8* SQL File 9* x

```

1 • Select c.CustomerName,t.i»¿TransactionID,t.TotalValue
2   from customers c
3  left join transactions t on c.i»¿CustomerID=t.CustomerID;
4

```

	CustomerName	i»¿TransactionID	TotalValue
▶	Lawrence Carroll	T00436	1300.92
	Lawrence Carroll	T00445	911.44
	Lawrence Carroll	T00085	614.94
	Lawrence Carroll	T00932	412.62
	Lawrence Carroll	T00015	114.6
	Elizabeth Lutz	T00675	254.72
	Elizabeth Lutz	T00148	770.74
	Elizabeth Lutz	T00129	382.76
	Elizabeth Lutz	T00155	454.52
	Michael Rivera	T00648	1385.2
	Michael Rivera	T00812	122.36

JOIN with product details:

Checking Customers Table SQL File 8* All customers (even without purc... SQL File 10* x SQL File 11* x SQLAddit

```

1 • Select t.i»¿TransactionID,c.CustomerName,p.ProductName,t.Quantity,t.TotalValue
2   from transactions t
3  join customers c on t.CustomerID=c.i»¿CustomerID
4  join products p on t.ProductID=p.i»¿ProductID;
5

```

	i»¿TransactionID	CustomerName	ProductName	Quantity	TotalValue
▶	T00436	Lawrence Carroll	TechPro Headphones	3	1300.92
	T00445	Lawrence Carroll	ActiveWear Smartwatch	2	911.44
	T00085	Lawrence Carroll	SoundWave Headphones	2	614.94
	T00932	Lawrence Carroll	HomeSense Wall Art	3	412.62
	T00015	Lawrence Carroll	SoundWave Cookbook	2	114.6
	T00675	Elizabeth Lutz	TechPro T-Shirt	2	254.72
	T00148	Elizabeth Lutz	ComfortLiving Sweater	2	770.74
	T00129	Elizabeth Lutz	BookWorld Rug	4	382.76
	T00155	Elizabeth Lutz	BookWorld Cookware Set	2	454.52
	T00648	Michael Rivera	ActiveWear Smartwatch	4	1385.2
	T00812	Michael Rivera	ActiveWear T-Shirt	4	122.36

Table: customers

Columns:

- i»¿CustomerID text
- CustomerName text
- Region text
- SignupDate text

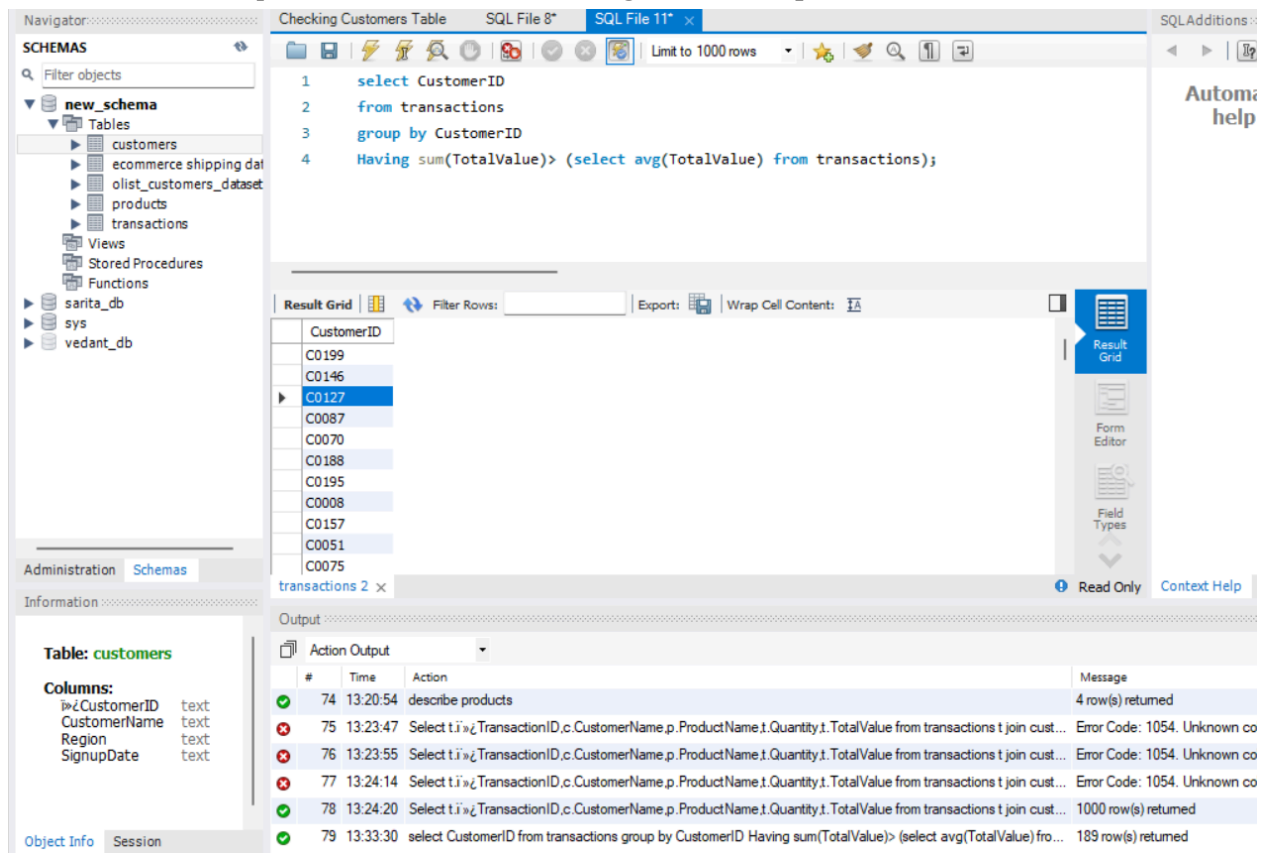
Object Info Session

Output

Action Output

#	Time	Action	Message
73	13:16:06	Select c.CustomerName,t.i»¿TransactionID,t.TotalValue from customers c left join transactions t on c.i»¿Custo...	1000 row(s) returned
74	13:20:54	describe products	4 row(s) returned
75	13:23:47	Select t.i»¿TransactionID,c.CustomerName,p.ProductName,t.Quantity,t.TotalValue from transactions t join cust...	Error Code: 1054. Unknow
76	13:23:55	Select t.i»¿TransactionID,c.CustomerName,p.ProductName,t.Quantity,t.TotalValue from transactions t join cust...	Error Code: 1054. Unknow
77	13:24:14	Select t.i»¿TransactionID,c.CustomerName,p.ProductName,t.Quantity,t.TotalValue from transactions t join cust...	Error Code: 1054. Unknow
78	13:24:20	Select t.i»¿TransactionID,c.CustomerName,p.ProductName,t.Quantity,t.TotalValue from transactions t join cust...	1000 row(s) returned

Customers who have purchased more than the **average total value per transaction**:



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

```
1 select CustomerID
2 from transactions
3 group by CustomerID
4 Having sum(TotalValue)> (select avg(TotalValue) from transactions);
```

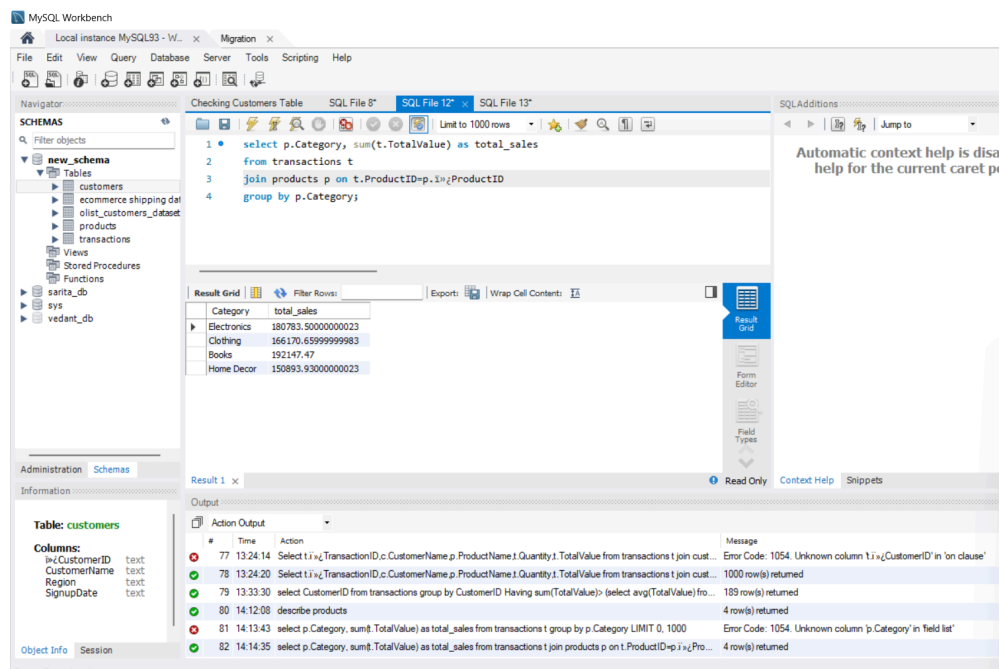
The Result Grid shows the following data:

CustomerID
C0199
C0146
C0127
C0087
C0070
C0188
C0195
C0008
C0157
C0051
C0075

The Output pane shows the execution log:

#	Time	Action	Message
74	13:20:54	describe products	4 row(s) returned
75	13:23:47	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	Error Code: 1054. Unknown co
76	13:23:55	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	Error Code: 1054. Unknown co
77	13:24:14	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	Error Code: 1054. Unknown co
78	13:24:20	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	1000 row(s) returned
79	13:33:30	select CustomerID from transactions group by CustomerID Having sum(TotalValue)> (select avg(TotalValue) fro...	189 row(s) returned

Total sales per product category using aggregate functions:



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

```
1 select p.Category, sum(t.TotalValue) as total_sales
2 from transactions t
3 join products p on t.ProductID=p.i.iProductID
4 group by p.Category;
```

The Result Grid shows the following data:

Category	total_sales
Electronics	180783.50000000023
Clothing	166170.65999999983
Books	192147.47
Home Decor	150893.93000000023

The Output pane shows the execution log:

#	Time	Action	Message
77	13:24:14	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	Error Code: 1054. Unknown column 'i.iCustomerID' in 'on clause'
78	13:24:20	Select t.i.iTransactionID.c.CustomerName.p.ProductName.t.Quantity.t.TotalValue from transactions t join cust...	1000 row(s) returned
79	13:33:30	select CustomerID from transactions group by CustomerID Having sum(TotalValue)> (select avg(TotalValue) fro...	189 row(s) returned
80	14:12:08	describe products	4 row(s) returned
81	14:13:43	select p.Category, sum(t.TotalValue) as total_sales from transactions t group by p.Category LIMIT 0, 1000	Error Code: 1054. Unknown column 'p.Category' in 'field list'
82	14:14:35	select p.Category, sum(t.TotalValue) as total_sales from transactions t join products p on t.ProductID=p.i.iPro...	4 row(s) returned

Average spend per customer using aggregate functions:

The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is as follows:

```
1 select c.CustomerName, avg(TotalValue) as avg_spent
2 from customers c
3 join transactions t on c.i_id=CustomerID = t.CustomerID
4 Group by c.CustomerName;
```

The result grid displays the following data:

CustomerName	avg_spent
Andrea Jenkins	494.82
Brittany Harvey	642.6999999999999
Kathryn Stevens	538.8133333333334
Travis Campbell	943.4614285714285
Timothy Perez	781.3725
Anna Ball	888.915
Jeremy McLaughlin	672.7116666666667
David Li	427.16100000000006
Miguel Wong	375.21999999999997
Robert Jones	601.36
Nicholas Ellis	814.2242857142858

The interface also shows a schema browser on the left with a tree view of the database structure, including tables, views, and functions. The bottom panel shows the output of the query, including a message indicating that 189 row(s) were returned.

View for customer purchase summary:

The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is as follows:

```
1 create view customer_purchase_summary_1 as
2 select c.i_id=CustomerID, c.CustomerName, sum(t.TotalValue) as total_spent
3 from customers c
4 join transactions t on c.i_id=CustomerID= t.CustomerID
5 group by c.i_id=CustomerID, c.CustomerName, c.Region;
```

The result grid displays the following data:

#	Time	Action	Message
86	14:30:46	select * from customer_purchase_summary LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP BY clause and contains n...
87	14:31:10	select * from customer_purchase_summary where total_spent>2000 LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP BY clause and contains n...
88	14:31:23	select * from customer_purchase_summary where total_spent>1000 LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP BY clause and contains n...
89	14:32:25	create view customer_purchase_summary as select c.i_id=CustomerID, c.CustomerName, sum(t.TotalValue) as t...	Error Code: 1050. Table 'customer_purchase_summary' already exists
90	14:33:19	create view customer_purchase_summary_1 as select c.i_id=CustomerID, c.CustomerName, sum(t.TotalValue) a...	0 row(s) affected
91	14:33:25	select * from customer_purchase_summary_1 where total_spent>1000 LIMIT 0, 1000	181 row(s) returned

The interface also shows a schema browser on the left with a tree view of the database structure, including tables, views, and functions. The bottom panel shows the output of the query, including a message indicating that 181 row(s) were returned.

View:

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the 'SCHEMAS' tree with 'new_schema' expanded, showing tables like 'customers', 'ecommerce_shipping_data', 'olist_customers_dataset', 'products', 'transactions', 'views', 'stored_procedures', 'functions', 'sys', and 'vedant_db'. The central pane shows a query: `select * from customer_purchase_summary_1 where total_spent>1000;`. The 'Result Grid' displays a list of customers with columns 'CustomerId', 'CustomerName', and 'total_spent'. The right pane shows 'SQLAdditions' with a message: 'Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.' Below the query, the 'Output' pane shows the 'Action Output' table.

#	Time	Action	Message
86	14:30:46	select * from customer_purchase_summary LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP B
87	14:31:10	select * from customer_purchase_summary where total_spent>2000 LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP B
88	14:31:23	select * from customer_purchase_summary where total_spent>1000 LIMIT 0, 1000	Error Code: 1055. Expression #1 of SELECT list is not in GROUP B
89	14:32:25	create view customer_purchase_summary as select c.i,c.CustomerID, c.CustomerName,sum(t.TotalValue) as t...	Error Code: 1050. Table 'customer_purchase_summary' already exists
90	14:33:19	create view customer_purchase_summary_1 as select c.i,c.CustomerID, c.CustomerName,sum(t.TotalValue) a...	0 row(s) affected
91	14:33:25	select * from customer_purchase_summary_1 where total_spent>1000 LIMIT 0, 1000	181 row(s) returned

Index Optimization:

- Customers_Region index for Customers table
- Transactions with CustomerID index in transactions table
- Product_Category for Products in transactions table

The screenshot shows the SQL Server Enterprise Manager interface with the 'SQL File 17*' tab active. The query editor contains three SQL statements: `CREATE INDEX index_customers_regions ON customers(Region(20));`, `CREATE INDEX idx_transactions_customer ON transactions(CustomerID(20));`, and `CREATE INDEX index_products_category on products(Category(20));`. The 'Output' pane shows the 'Action Output' table with details for each index creation.

#	Time	Action	Message	Duration / Fetch
93	14:38:13	CREATE INDEX index_customers_regions ON customers(Region)	Error Code: 1170. BLOB/TEXT column 'Region' used in key specification without a key length	0.000 sec
94	14:38:54	CREATE INDEX index_customers_regions ON customers(Region(20))	0 row(s) affected. Records: 0 Duplicates: 0 Warnings: 0	0.078 sec
95	14:40:05	CREATE INDEX index_customers_regions ON customers(Region(20))	Error Code: 1061. Duplicate key name 'index_customers_regions'	0.016 sec
96	14:40:19	CREATE INDEX idx_transactions_customer ON transactions(CustomerID(20))	0 row(s) affected. Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
97	14:40:44	CREATE INDEX index_transactions_customer ON transactions(CustomerID(20))	0 row(s) affected. 1 warning(s): 1831 Duplicate index 'index_transactions_customer' defined on the table 'new_...	0.031 sec
98	14:41:40	CREATE INDEX index_products_category on products(Category(20))	0 row(s) affected. Records: 0 Duplicates: 0 Warnings: 0	0.047 sec

To check if the indexes are made:

Navigation

SCHEMAS

Filter objects

new_schema

Tables

customers

ecommerce_shipping_data

olist_customers_dataset

products

transactions

Views

Stored Procedures

Functions

sys

vedant_db

Checking Customers Table

SQL File 8*

SQL File 13*

Index oPTIMIZATION

SQL File 18*

1 SHOW INDEX FROM transactions;

2

Result Grid

Filter Rows:

Export:

Wrap Cell Contents:

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub
transactions	1	idx_transactions_customer	1	CustomerID	A	199	20
transactions	1	index_transactions_customer	1	CustomerID	A	199	20

Result Grid

Form Editor

Field Types

Administration

Schemas

Information

Table: customers

Columns:

idx_CustomerID text

CustomerName text

Region text

SignupDate text

Object Info

Session

Result 1

Read Only

Cont

Output

Action Output

#	Time	Action	Message
94	14:38:54	CREATE INDEX index_customers_regions ON customers(Region(20))	0 row(s) affected
95	14:40:05	CREATE INDEX index_customers_regions ON customers(Region(20))	Error Code: 1061. D
96	14:40:19	CREATE INDEX idx_transactions_customer ON transactions(CustomerID(20))	0 row(s) affected
97	14:40:44	CREATE INDEX index_transactions_customer ON transactions(CustomerID(20))	0 row(s) affected, 1
98	14:41:40	CREATE INDEX index_products_category on products(Category(20))	0 row(s) affected
99	14:45:13	SHOW INDEX FROM transactions	2 row(s) returned