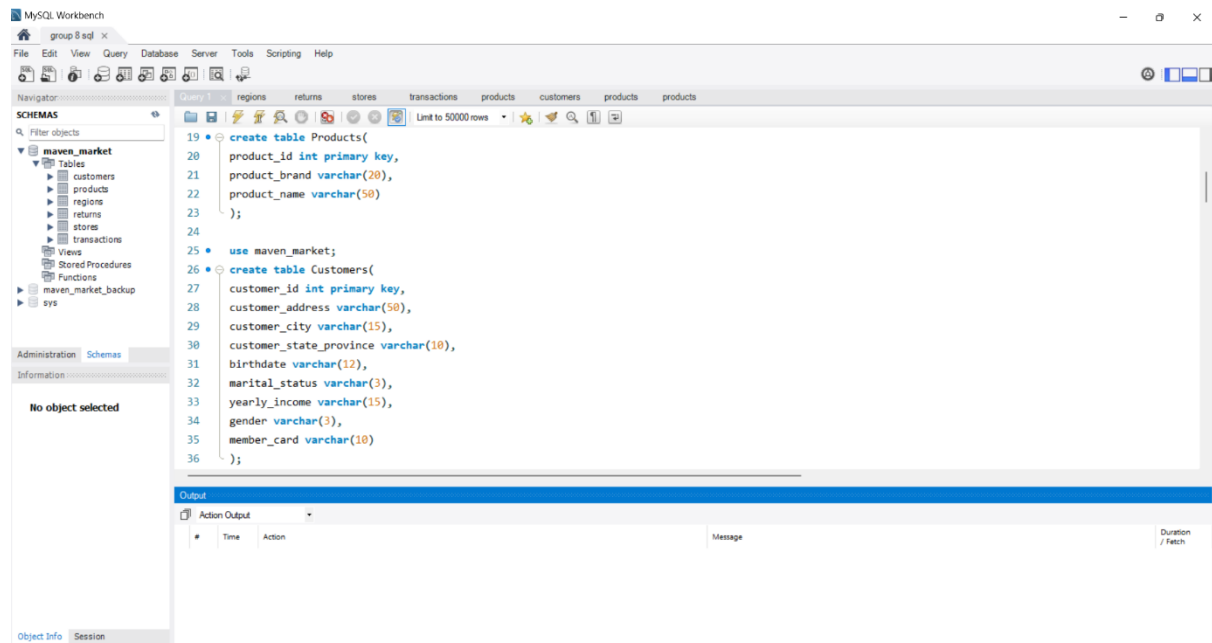
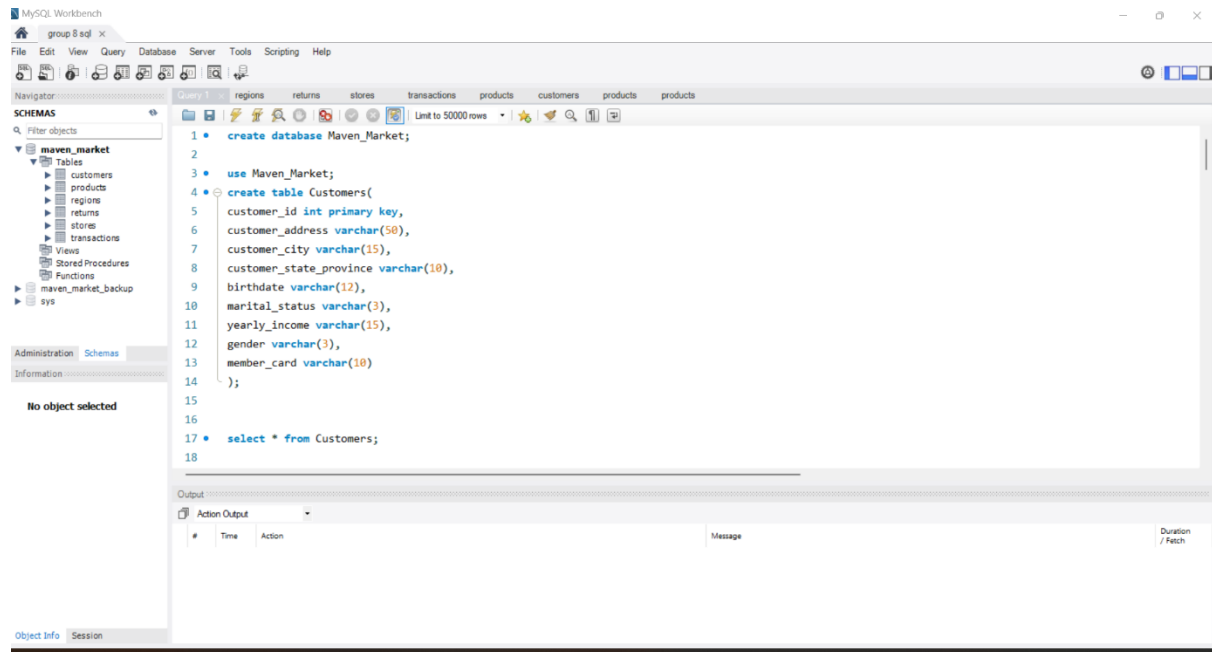
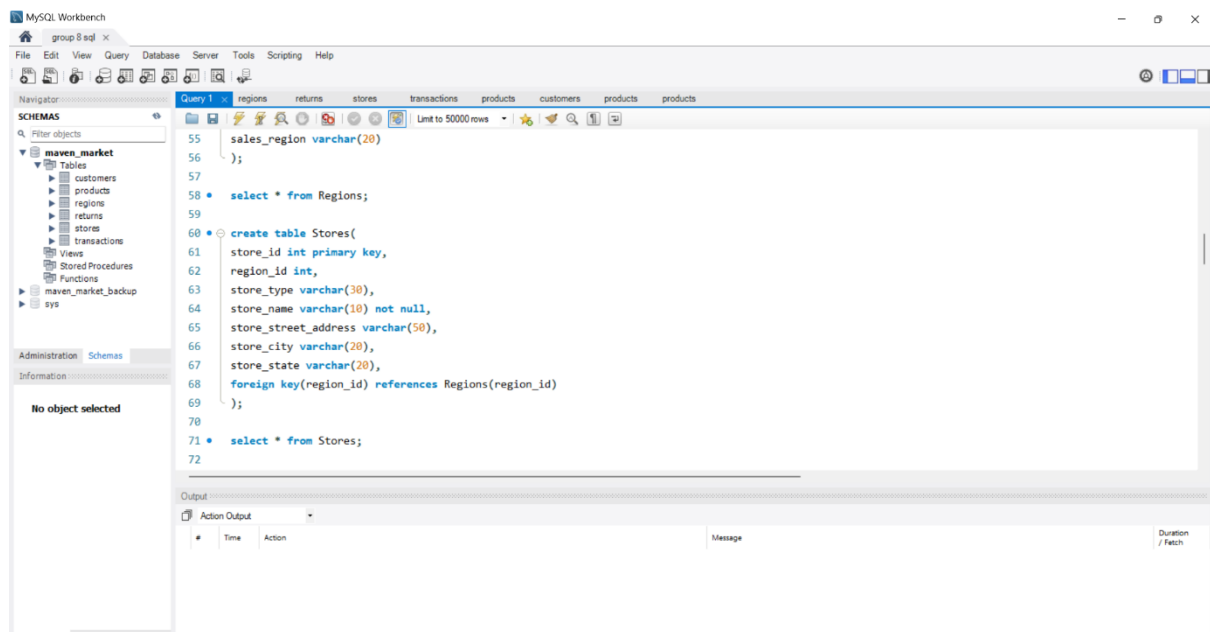
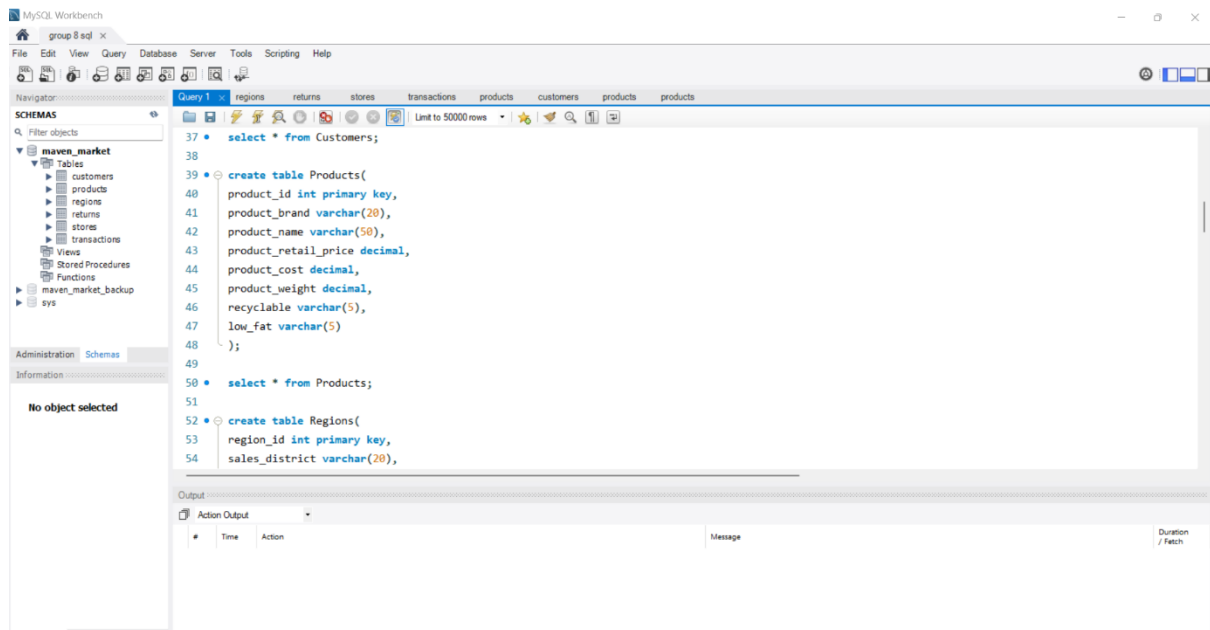
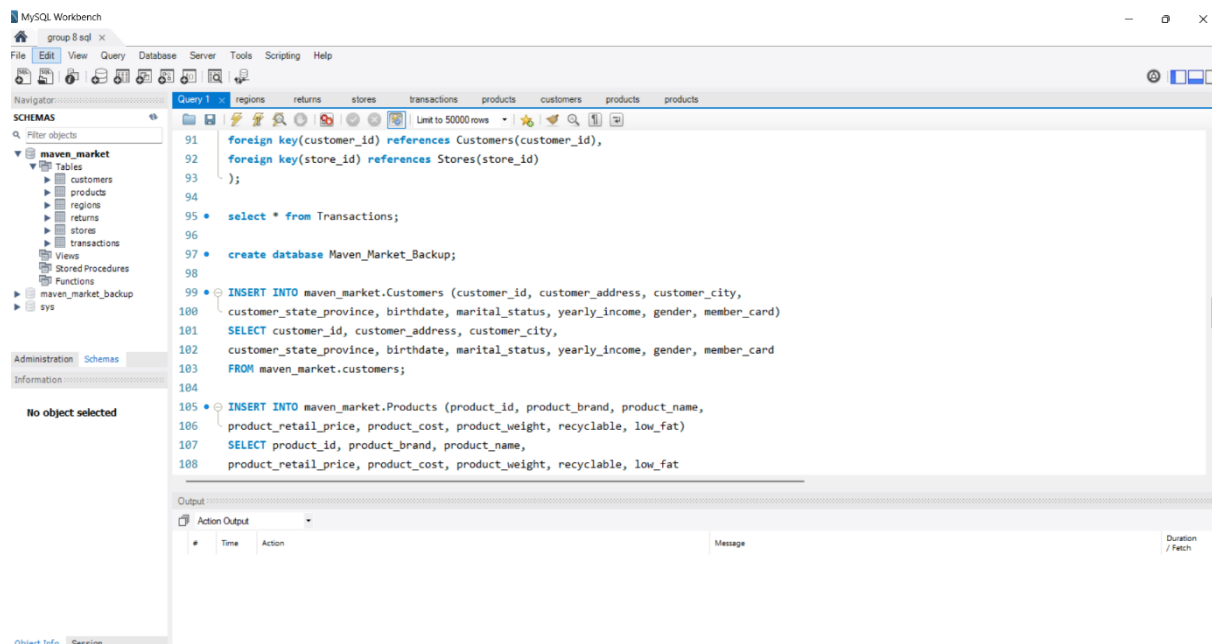
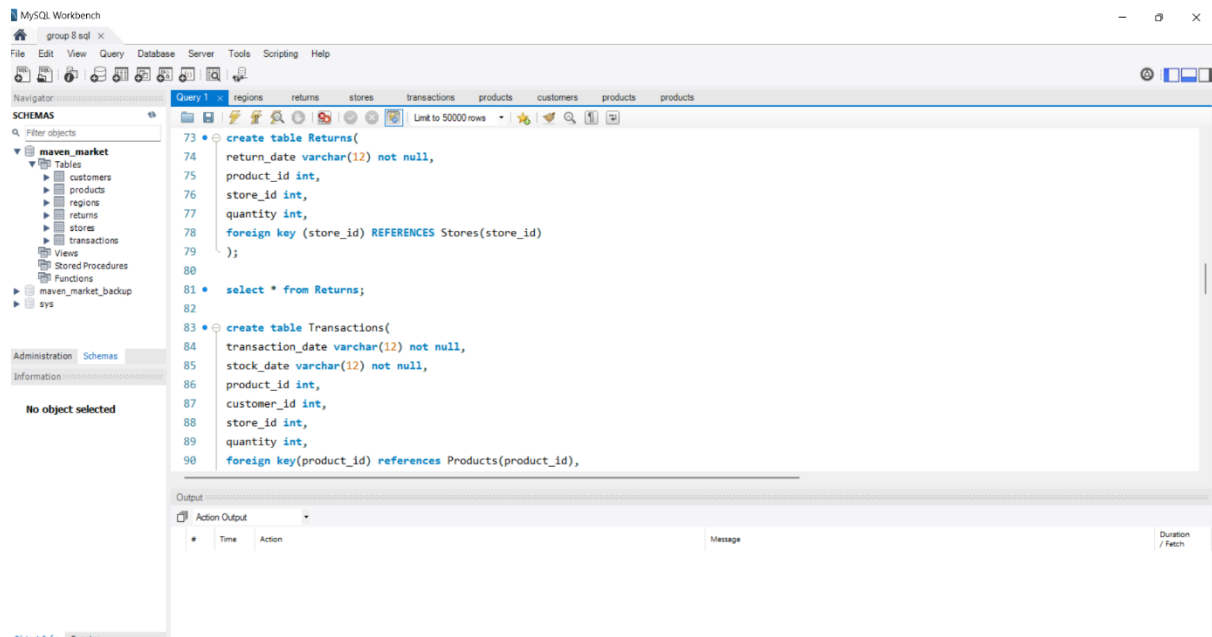








MySQL Workbench

group 8 sql

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

maven_market

Tables

- customers
- products
- regions
- returns
- stores
- transactions

Views

Stored Procedures

Functions

maven_market_backup

sys

Administration Schemas

Information

No object selected

Object Info Session

Query 1

regions returns stores transactions products customers products products

Limit to 50000 rows

```

109 FROM maven_market.products;
110
111 • INSERT INTO maven_market.regions (region_id, sales_district, sales_region)
112   SELECT region_id, sales_district, sales_region
113   FROM maven_market.regions;
114
115 • INSERT INTO maven_market.stores (store_id, region_id, store_type, store_name, store_street_address, store_city, store_state)
116   SELECT store_id, region_id, store_type, store_name, store_street_address, store_city, store_state
117   FROM maven_market.stores;
118
119 • INSERT INTO maven_market.returns (return_date, product_id, store_id, quantity)
120   SELECT return_date, product_id, store_id, quantity
121   FROM maven_market.returns;
122
123 • INSERT INTO maven_market.transactions (transaction_date, stock_date, product_id, customer_id, store_id, quantity)
124   SELECT transaction_date, stock_date, product_id, customer_id, store_id, quantity
125   FROM maven_market.transactions;
126

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
---	------	--------	---------	------------------

MySQL Workbench

group 8 sql

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHMAS

Filter objects

maven_market

Tables

- customers
- products
- regions
- returns
- stores
- transactions

Views

Stored Procedures

Functions

maven_market_backup

sys

Administration Schemas

Information

No object selected

Object Info Session

Create a new stored procedure in the active schema in the connected server

products customers products products

Limit to 50000 rows

```

127
128 #1. Average cost per product.
129 • SELECT product_name, AVG(product_cost) AS average_cost
130   FROM maven_market.products
131   GROUP BY product_name;
132
133 #2. Store-wise sales.
134 • SELECT store_name, SUM(quantity) AS total_sales
135   FROM maven_market.transactions
136   INNER JOIN stores ON transactions.store_id = stores.store_id
137   GROUP BY store_name;
138
139 #3. Number of customers by marital status.
140 • SELECT customers.marital_status, COUNT(DISTINCT transactions.customer_id) AS customer_count
141   FROM maven_market.transactions
142   INNER JOIN customers ON transactions.customer_id = customers.customer_id
143   GROUP BY customers.marital_status;
144

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
---	------	--------	---------	------------------

MySQL Workbench

group 8 sql

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

maven_market

- Tables
 - customers
 - products
 - regions
 - returns
 - stores
 - transactions
- Views
- Stored Procedures
- Functions
- maven_market_backup
- sys

Administration Schemas

Information

No object selected

Object Info Session

Query 1

regions returns stores transactions products customers products products

Limit to 50000 rows

```
145 #4. Total revenue.
146 • SELECT SUM(quantity*product_retail_price) AS total_revenue
147 FROM maven_market.transactions
148 INNER JOIN products ON Transactions.product_id = products.product_id;
149
150 #5. Average weight per products.
151 • SELECT product_name,AVG(product_weight) AS avg_weight
152 FROM Products
153 GROUP BY product_name;
154
155 #6. Total revenue by region.
156 • SELECT regions.sales_region,SUM(transactions.quantity*products.product_retail_price) AS total_revenue
157 FROM Transactions
158 INNER JOIN products ON transactions.product_id = products.product_id
159 INNER JOIN stores ON transactions.store_id = stores.store_idcustomerscustomerscustomerscustomers
160 INNER JOIN regions ON stores.region_id = regions.region_id
161 GROUP BY regions.sales_region;
162
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
---	------	--------	---------	------------------

MySQL Workbench

group 8 sql

File Edit View Query Database Server Tools Scripting Help

Navigator: regions returns stores transactions products customers products products

SCHEMAS

Filter objects

maven_market

Tables

customers

products

regions

returns

stores

transactions

Views

Stored Procedures

Functions

maven_market_backup

sys

Administration Schemas

Information

No object selected

Query 1: regions returns stores transactions products customers products products

Limit to 50000 rows

```

163 #7. Top returned products.
164 • SELECT products.product_name,SUM(returns.quantity) AS return_quantity
165 FROM Returns
166 INNER JOIN products ON returns.product_id = products.product_id
167 GROUP BY product_name;
168
169 #8. Revenue by customers income.
170 • SELECT Customers.yearly_income,SUM(transactions.quantity*products.product_retail_price) AS total_revenue
171 FROM Transactions
172 INNER JOIN products ON transactions.product_id = products.product_id
173 INNER JOIN customers ON transactions.customer_id = customers.customer_id
174 GROUP BY yearly_income;
175
176 #9. Total number of transactions.
177 • SELECT COUNT(*) AS total_transactions
178 FROM Transactions;
179
180 #10. Revenue by customer membership.

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
---	------	--------	---------	------------------

MySQL Workbench

group 8 sql

File Edit View Query Database Server Tools Scripting Help

Navigator: regions returns stores transactions products customers products products

SCHEMAS

Filter objects

maven_market

Tables

customers

products

regions

returns

stores

transactions

Views

Stored Procedures

Functions

maven_market_backup

sys

Administration Schemas

Information

No object selected

Query 1: regions returns stores transactions products customers products products

Limit to 50000 rows

```

169 #8. Revenue by customers income.
170 • SELECT Customers.yearly_income,SUM(transactions.quantity*products.product_retail_price) AS total_revenue
171 FROM Transactions
172 INNER JOIN products ON transactions.product_id = products.product_id
173 INNER JOIN customers ON transactions.customer_id = customers.customer_id
174 GROUP BY yearly_income;
175
176 #9. Total number of transactions.
177 • SELECT COUNT(*) AS total_transactions
178 FROM Transactions;
179
180 #10. Revenue by customer membership.
181 • SELECT customers.member_card,SUM(transactions.quantity*products.product_retail_price) AS total_revenue
182 FROM Transactions
183 INNER JOIN products ON transactions.product_id = products.product_id
184 INNER JOIN customers ON transactions.customer_id = customers.customer_id
185 GROUP BY member_card;
186

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
---	------	--------	---------	------------------