



Lanka Nippon Biztech Institute

DATABASE FINAL PROJECT

Course code – ICT 1403

Course Name – Database II

Name : K.Minori Sanjana Perera

Index Number: UoG03 22051

Batch : UoG 03 Group 02

Contents

1. Design and implementation of the database.

1.1. Provide an analysis of entities, attributes and relationships with cardinalities of the given scenario and draw an ER diagram by including cardinality constraints.

1.2. Translate the ER diagram into relational schemas by showing the relation with each through primary keys and foreign keys.

1.3. Design a relational database system according to the ER diagram you have created (Use MySQL DDL statements. Primary keys and foreign key/s are compulsory in implementation).

2. Use of querying tools. Perform following MySQL queries using the tables that have created in task 1.2. Provide evidence

For MySQL queries and results.

2.1. Retrieve the details of a customer's order, including the clothing items ordered, their sizes, and the delivery status.

2.2. Delete a specific customer's order along with all the associated order details, such as the clothing items and delivery information.

2.3. Update the customer's order status to "Cancelled" if they decide to cancel the order before delivery.

2.4. Create a stored procedure to find the customer details and the order details who purchased any product from a given category (Input category from the keyboard).

2.5. Create a view to find the supplier details, product and the stock details as a summary.

2.6. Create a trigger that automatically updates the stock quantity of clothing items according to the order payment status.

2.7. Create two roles as 'database admin' and 'stock keeper' for the database. Create users under 'database admin' and 'stock keeper'. Grant following privileges to users under each role.

- Database admin - Grant full access to all tables, allowing users assigned to this role to access and modify any table within the database.
- Stock keeper - Grant CRUD (Create, Read, Update, Delete) operations only for the stocks, ensuring that users in this role can only perform actions related to managing stock data.

INTRODUCTION

A SHORT INTRODUCTION FOR WHAT DATABASE SIMPLY IS;

A database is an organized collection of data that can be easily accessed, managed, and updated. Database management systems (DBMS) are software systems that are used to manage databases. DBMS helps users and programmers to interact with databases, retrieve information and perform operations on data stored in them. Database assignments are given to students who are learning about database management and design. A database assignment can vary in complexity, but it typically involves tasks such as designing a database schema, creating tables, inserting data, and running queries to retrieve information from the database. In some cases, students may be asked to optimize database performance or troubleshoot issues with an existing database.

HERE, IN THIS ASSIGNMENT I WILL FOCUS ON,

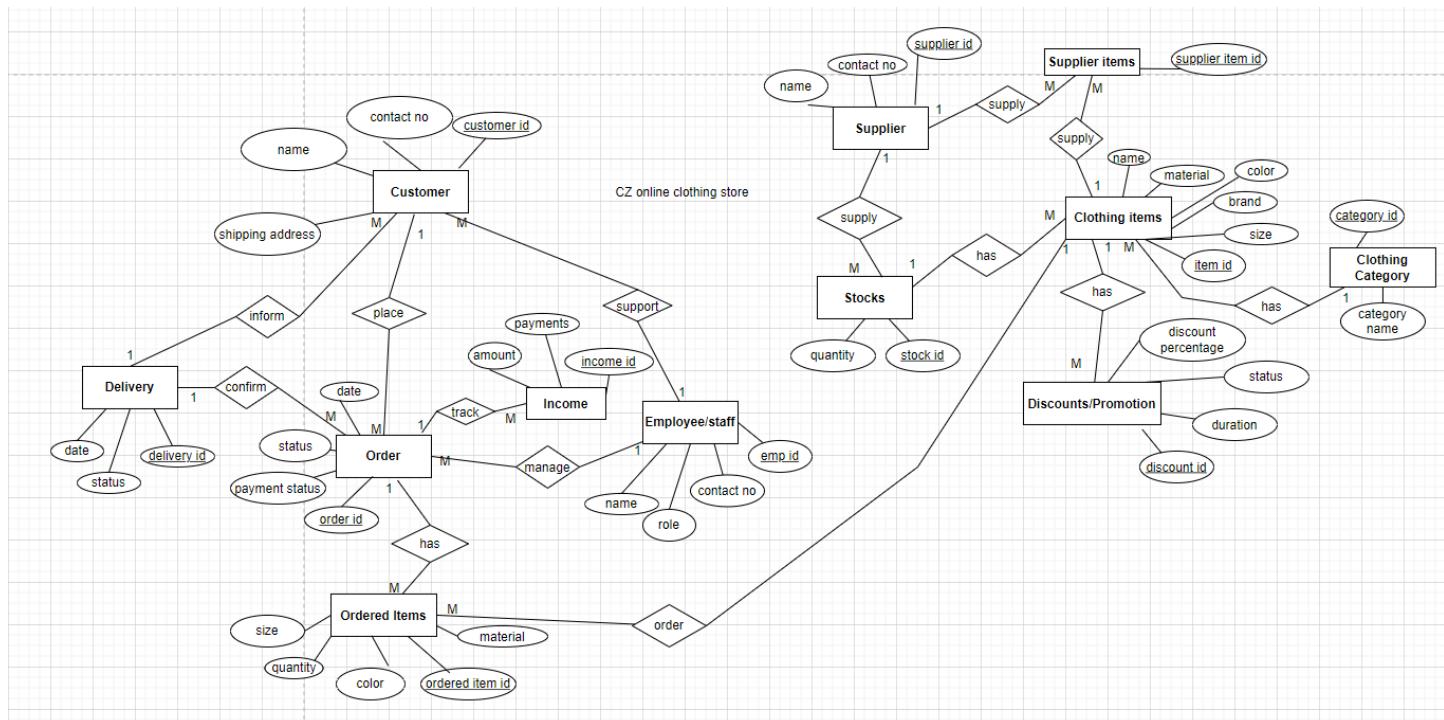
- Understanding data models and database technologies
- Be able to design and implement relational database systems
- Be able to use manipulation and querying tools
- Be able to develop a secure relational database system
- Grant privileges to users
- Grant full access to tables
- Creating users

Case Study

The CZ online clothing store is in need of developing a database system to manage their customer details, orders, stocks and supplies and delivery information. Their main goal is to ensure a smooth and satisfying shopping experience for customers. The following details are activities of their requirements provided regarding the activities of their requirements.

1. Design and implementation of the database.

1.1. Provide an analysis of entities, attributes and relationships with cardinalities of the given scenario and draw an ER diagram by including cardinality constraints.



ASSUMPTIONS-

- ✓ A clothing category has many clothing items and a clothing item has only one clothing category.
- ✓ A staff member supports many Customers while a customer has only one staff member at a time.
- ✓ A stock has many clothing items and a clothing item is given by a single stock.
- ✓ Many customers are informed about the delivery information by the Store's delivery person.

1.2. Translate the ER diagram into relational schemas by showing the relation with each through primary keys and foreign keys.

1. Customers

<u>Customer id</u>	name	Contact no	Shipping address
--------------------	------	------------	------------------

2. Orders

<u>Order id</u>	<u>Customer id</u>	status	Payment status	Date
-----------------	--------------------	--------	----------------	------

3. Ordered Items

<u>Ordered item id</u>	<u>Order id</u>	<u>Item id</u>	color	material	quantity	Size
------------------------	-----------------	----------------	-------	----------	----------	------

4. Delivery

<u>Delivery id</u>	<u>Order id</u>	<u>Customer id</u>	status	date
--------------------	-----------------	--------------------	--------	------

5. Income

<u>Income id</u>	<u>Order id</u>	payments	amount
------------------	-----------------	----------	--------

6. Employees/Staff

<u>Emp id</u>	<u>Order id</u>	<u>Customer id</u>	name	role	Contact no
---------------	-----------------	--------------------	------	------	------------

7. Clothing items

<u>Item id</u>	<u>Category id</u>	name	materials	colors	sizes	brands
----------------	--------------------	------	-----------	--------	-------	--------

8. Clothing category

<u>Category id</u>	Category name
--------------------	---------------

9. Suppliers

<u>Supplier id</u>	name	Contact no
--------------------	------	------------

10. Supplier Items

<u>Supplier item id</u>	<u>Supplier id</u>	<u>Item id</u>
-------------------------	--------------------	----------------

11. Stock

<u>Stock id</u>	<u>Item id</u>	quantity
-----------------	----------------	----------

12. Discounts/Promotions

<u>Discount percentage</u>	<u>Item id</u>	duration	status
----------------------------	----------------	----------	--------

1.3. Design a relational database system according to the ER diagram you have created (Use MySQL DDL statements. Primary keys and foreign key/s are compulsory in implementation).

```

MySQL Workbench - Local instance wampmysqld54
File Edit View Query Database Server Tools Scripting Help
Navigator: labsheet* dbproject2*
1 • create database czonlinelistingstore;
2 • use czonlinelistingstore;
3
4 • CREATE TABLE ClothingCategory (
5     categoryid INT PRIMARY KEY,
6     categoryname VARCHAR(255)
7 );
8
9 • CREATE TABLE ClothingItems (
10    itemid INT PRIMARY KEY,
11    itemname VARCHAR(255),
12    materials VARCHAR(255),
13    colors VARCHAR(255),
14    sizes VARCHAR(255),
15    brands VARCHAR(255),
16    categoryid INT,
17    FOREIGN KEY (categoryid) REFERENCES ClothingCategory(categoryid)
18 );
19
20 • CREATE TABLE Suppliers (

```

No object selected

Output:

#	Time	Action	Message	Duration / Fetch
1	18:44:34	create database czonlinelistingstore	1 row(s) affected	0.015 sec
2	18:45:18	use czonlinelistingstore	0 row(s) affected	0.000 sec
3	18:52:09	CREATE TABLE ClothingCategory (categoryid INT PRIMARY KEY, categoryname VARCHAR(255))	0 row(s) affected	0.016 sec
4	19:16:55	CREATE TABLE ClothingItems (itemid INT PRIMARY KEY, itemname VARCHAR(255), materials VARCHAR(255), colors VARCHAR(255), sizes VARCHAR(255), brands VARCHAR(255), categoryid INT, FOREIGN KEY (categoryid) REFERENCES ClothingCategory(categoryid))	0 row(s) affected	0.016 sec
5	19:17:20	CREATE TABLE Suppliers (supplierid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255))	0 row(s) affected	0.016 sec
6	19:17:20	CREATE TABLE Suppliers (supplierid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255))	0 row(s) affected	0.016 sec

Object Info Session

Query Completed

```

MySQL Workbench - Local instance wampmysqld54
File Edit View Query Database Server Tools Scripting Help
Navigator: labsheet* dbproject2*
28 • CREATE TABLE Suppliers (
29    supplierid INT PRIMARY KEY,
30    name VARCHAR(255),
31    contactno VARCHAR(255)
32 );
33
34 • CREATE TABLE SupplierItems (
35    supplieritemid INT PRIMARY KEY,
36    supplierid INT,
37    itemid INT,
38    FOREIGN KEY (supplierid) REFERENCES Suppliers(supplierid),
39    FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid)
40 );
41
42 • CREATE TABLE Stock (
43    stockid INT PRIMARY KEY,
44    quantity INT,
45    itemid INT,
46    FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid)
47 );
48
49 • UPDATE Suppliers SET contactno = '0123456789' WHERE supplierid = 1;

```

No object selected

Output:

#	Time	Action	Message	Duration / Fetch
1	18:44:34	create database czonlinelistingstore	1 row(s) affected	0.015 sec
2	18:45:18	use czonlinelistingstore	0 row(s) affected	0.000 sec
3	18:52:09	CREATE TABLE ClothingCategory (categoryid INT PRIMARY KEY, categoryname VARCHAR(255))	0 row(s) affected	0.016 sec
4	19:16:55	CREATE TABLE ClothingItems (itemid INT PRIMARY KEY, itemname VARCHAR(255), materials VARCHAR(255), colors VARCHAR(255), sizes VARCHAR(255), brands VARCHAR(255), categoryid INT, FOREIGN KEY (categoryid) REFERENCES ClothingCategory(categoryid))	0 row(s) affected	0.016 sec
5	19:17:20	CREATE TABLE Suppliers (supplierid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255))	0 row(s) affected	0.016 sec
6	19:17:20	CREATE TABLE Suppliers (supplierid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255))	0 row(s) affected	0.016 sec
7	19:17:20	CREATE TABLE Suppliers (supplierid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255))	0 row(s) affected	0.016 sec
8	19:17:20	CREATE TABLE SupplierItems (supplieritemid INT PRIMARY KEY, supplierid INT, itemid INT, FOREIGN KEY (supplierid) REFERENCES Suppliers(supplierid), FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid))	0 row(s) affected	0.016 sec
9	19:17:20	CREATE TABLE Stock (stockid INT PRIMARY KEY, quantity INT, itemid INT, FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid))	0 row(s) affected	0.016 sec
10	19:17:20	UPDATE Suppliers SET contactno = '0123456789' WHERE supplierid = 1;	1 row(s) affected	0.016 sec

Object Info Session

Query Completed

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet dbproject2

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
7	19:17:32	CREATE TABLE Stock (stockid INT PRIMARY KEY, quantity INT, itemid INT, FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid))	0 row(s) affected	0.016 sec
8	19:17:51	CREATE TABLE Discounts (discountid INT PRIMARY KEY, discountpercentage DECIMAL(5, 2), duration VARCHAR(255), status VARCHAR(255), itemid INT, FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid))	0 row(s) affected	0.031 sec
9	19:17:58	CREATE TABLE Customers (customerid INT PRIMARY KEY, name VARCHAR(255), contactno VARCHAR(255), shippingaddr VARCHAR(255))	0 row(s) affected	0.016 sec
10	19:18:04	CREATE TABLE Orders (orderid INT PRIMARY KEY, orderstatus VARCHAR(255), paymentstatus VARCHAR(255), deliverydate DATE, customerid INT, FOREIGN KEY (customerid) REFERENCES Customers(customerid))	0 row(s) affected	0.015 sec
11	19:18:12	CREATE TABLE OrderedItems (ordereditemid INT PRIMARY KEY, size VARCHAR(255), color VARCHAR(255), material VARCHAR(255), quantity INT, orderid INT, itemid INT, FOREIGN KEY (orderid) REFERENCES Orders(orderid), FOREIGN KEY (itemid) REFERENCES ClothingItems(itemid))	0 row(s) affected	0.032 sec
12	19:18:17	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate DATE, customerid INT, FOREIGN KEY (customerid) REFERENCES Customers(customerid))	Error Code: 1072. Key column 'customerid' doesn't exist in table	0.015 sec

Object Info Session

Query Completed

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet dbproject2

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

Output:

Action Output

#	Time	Action	Message	Duration / Fetch
12	19:18:17	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate DATE, customerid INT, FOREIGN KEY (customerid) REFERENCES Customers(customerid))	Error Code: 1072. Key column 'customerid' doesn't exist in table	0.015 sec
13	19:18:21	CREATE TABLE Income (incometid INT PRIMARY KEY, payments VARCHAR(255), amount DECIMAL(10, 2))	0 row(s) affected	0.016 sec
14	19:18:33	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	Error Code: 1072. Key column 'orderid' doesn't exist in table	0.000 sec
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate DATE, customerid INT, FOREIGN KEY (customerid) REFERENCES Customers(customerid))	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	0 row(s) affected	0.016 sec

Object Info Session

Query Completed

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet* dbproject2*

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

```

78 • CREATE TABLE Delivery (
  deliveryid INT PRIMARY KEY,
  deliverystatus VARCHAR(255),
  deliverydate DATE,
  orderid INT,
  customerid INT,
  FOREIGN KEY (orderid) REFERENCES Orders(orderid),
  FOREIGN KEY (customerid) REFERENCES Customers(customerid)
);

88 • CREATE TABLE Income (
  incom eid INT PRIMARY KEY,
  payments VARCHAR(255),
  amount DECIMAL(10, 2),
  orderid INT,
  FOREIGN KEY (orderid) REFERENCES Orders(orderid)
);

96 • CREATE TABLE Employees (
  empid INT PRIMARY KEY,
  ...
);

```

Output:

#	Time	Action	Message	Duration / Fetch
12	19:18:17	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate ...)	Error Code: 1072. Key column 'customerid' doesn't exist in table	0.015 sec
13	19:18:21	CREATE TABLE Income (incom eid INT PRIMARY KEY, payments VARCHAR(255), amount DECIMAL(10, 2))	0 row(s) affected	0.016 sec
14	19:18:33	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	Error Code: 1072. Key column 'orderid' doesn't exist in table	0.000 sec
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate ...)	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	0 row(s) affected	0.016 sec

Object Info Session

Query Completed

7:24 PM 8/10/2023

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet* dbproject2*

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

```

86 );
87
88 • CREATE TABLE Income (
  incom eid INT PRIMARY KEY,
  payments VARCHAR(255),
  amount DECIMAL(10, 2),
  orderid INT,
  FOREIGN KEY (orderid) REFERENCES Orders(orderid)
);

96 • CREATE TABLE Employees (
  empid INT PRIMARY KEY,
  name VARCHAR(255),
  role VARCHAR(255),
  contactno VARCHAR(255),
  orderid INT,
  customerid INT,
  FOREIGN KEY (orderid) REFERENCES Orders(orderid),
  FOREIGN KEY (customerid) REFERENCES Customers(customerid)
);


```

Output:

#	Time	Action	Message	Duration / Fetch
12	19:18:17	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate ...)	Error Code: 1072. Key column 'customerid' doesn't exist in table	0.015 sec
13	19:18:21	CREATE TABLE Income (incom eid INT PRIMARY KEY, payments VARCHAR(255), amount DECIMAL(10, 2))	0 row(s) affected	0.016 sec
14	19:18:33	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	Error Code: 1072. Key column 'orderid' doesn't exist in table	0.000 sec
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate ...)	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255), ...)	0 row(s) affected	0.016 sec

Object Info Session

Query Completed

7:24 PM 8/10/2023

MySQL Workbench Local instance wampmysqld64

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

No object selected

Customers 1 x

Output:

#	Time	Action	Message	Duration / Fetch
13	19:18:21	CREATE TABLE Income (incomeid INT PRIMARY KEY, payments VARCHAR(255), amount DECIMAL(10,2))	0 row(s) affected	0.016 sec
14	19:18:33	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255))	Error Code: 1072. Key column 'ordid' doesn't exist in table	0.000 sec
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate DATE)	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255))	0 row(s) affected	0.016 sec
17	19:26:22	select * from Customers LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
18	19:26:22	select * from Orders LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

MySQL Workbench Local instance wampmysqld64

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

No object selected

Orders 2 x

Output:

#	Time	Action	Message	Duration / Fetch
14	19:18:33	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255))	Error Code: 1072. Key column 'ordid' doesn't exist in table	0.000 sec
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate DATE)	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255))	0 row(s) affected	0.016 sec
17	19:26:22	select * from Customers LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
18	19:26:54	select * from Orders LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet dbproject2

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

OrderedItems 3

Output:

#	Time	Action	Message	Duration / Fetch
15	19:21:45	CREATE TABLE Delivery (deliveryid INT PRIMARY KEY, deliverystatus VARCHAR(255), deliverydate ...	0 row(s) affected	0.016 sec
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255)...	0 row(s) affected	0.016 sec
17	19:26:22	select * from Customers LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
18	19:26:54	select * from Orders LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
19	19:27:06	select * from OrderedItems LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

MySQL Workbench

Local instance wampmysqld64

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet dbproject2

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

ClothingItems 4

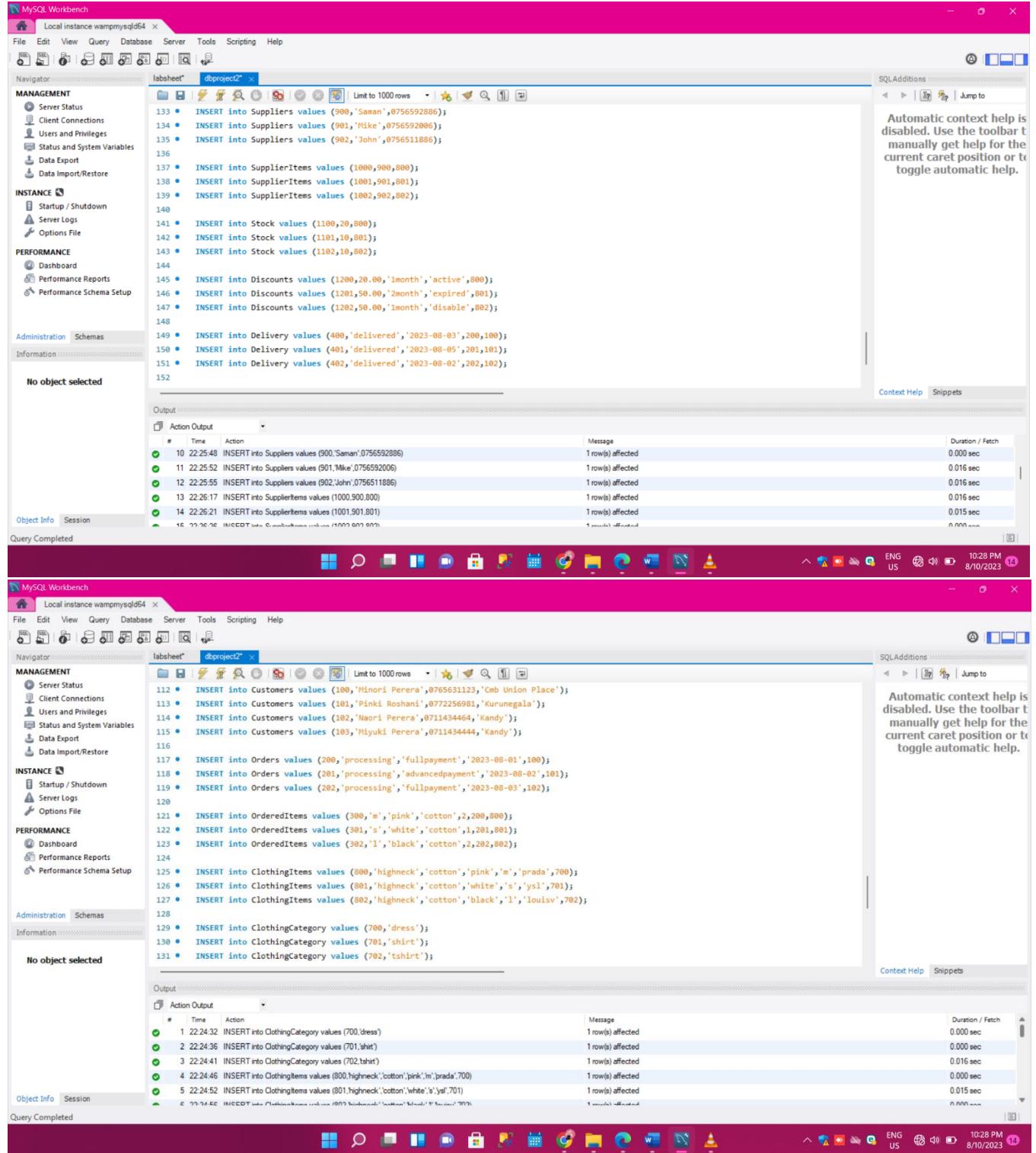
Output:

#	Time	Action	Message	Duration / Fetch
16	19:21:50	CREATE TABLE Employees (empid INT PRIMARY KEY, name VARCHAR(255), role VARCHAR(255)...	0 row(s) affected	0.016 sec
17	19:26:22	select * from Customers LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
18	19:26:54	select * from Orders LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
19	19:27:06	select * from OrderedItems LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
20	19:27:19	select * from ClothingItems LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

Query Completed

- Inserting values to the table in the database.



The screenshot displays two separate sessions in MySQL Workbench, both titled "dbproject2".

Session 1 (Top): Inserting into Suppliers

```

133 • INSERT into Suppliers values (900,'Saman',0756592886);
134 • INSERT into Suppliers values (901,'Mike',0756592006);
135 • INSERT into Suppliers values (902,'John',0756511886);
136
137 • INSERT into SupplierItems values (1000,900,800);
138 • INSERT into SupplierItems values (1001,901,801);
139 • INSERT into SupplierItems values (1002,902,802);
140
141 • INSERT into Stock values (1100,20,800);
142 • INSERT into Stock values (1101,10,801);
143 • INSERT into Stock values (1102,10,802);
144
145 • INSERT into Discounts values (1200,20.00,'1month','active',800);
146 • INSERT into Discounts values (1201,50.00,'2month','expired',801);
147 • INSERT into Discounts values (1202,50.00,'1month','disabled',802);
148
149 • INSERT into Delivery values (400,'delivered','2023-08-03',200,100);
150 • INSERT into Delivery values (401,'delivered','2023-08-05',201,101);
151 • INSERT into Delivery values (402,'delivered','2023-08-02',202,102);
152

```

Session 2 (Bottom): Inserting into Customers

```

112 • INSERT into Customers values (100,'Minori Perera',0765631123,'Cmb Union Place');
113 • INSERT into Customers values (101,'Pinki Roshani',0772256981,'Kurunegala');
114 • INSERT into Customers values (102,'Naori Perera',0711434464,'Kandy');
115 • INSERT into Customers values (103,'Miyuki Perera',0711434444,'Kandy');
116
117 • INSERT into Orders values (200,'processing','fullpayment','2023-08-01',100);
118 • INSERT into Orders values (201,'processing','advancedpayment','2023-08-02',101);
119 • INSERT into Orders values (202,'processing','fullpayment','2023-08-03',102);
120
121 • INSERT into OrderedItems values (300,'m','pink','cotton',2,200,800);
122 • INSERT into OrderedItems values (301,'s','white','cotton',1,201,801);
123 • INSERT into OrderedItems values (302,'l','black','cotton',2,202,802);
124
125 • INSERT into ClothingItems values (800,'highneck','cotton','pink','m','prada',700);
126 • INSERT into ClothingItems values (801,'highneck','cotton','white','s','ysl',701);
127 • INSERT into ClothingItems values (802,'highneck','cotton','black','l','louisv',702);
128
129 • INSERT into ClothingCategory values (700,'dress');
130 • INSERT into ClothingCategory values (701,'shirt');
131 • INSERT into ClothingCategory values (702,'tshirt');
132

```

Both sessions show successful insertions with 1 row(s) affected and a duration of 0.000 sec.

The screenshot shows the MySQL Workbench interface. In the top-left, there's a 'Navigator' panel with sections like MANAGEMENT, INSTANCE, PERFORMANCE, and Administration. The central area is a 'labsheet' tab containing a script editor with several SQL INSERT statements. Below the script editor is an 'Output' pane showing the results of the executed queries, including timestamps, actions, and duration/fetch times. The bottom status bar indicates the session was completed at 10:28 PM on 8/10/2023.

```

140
141 • INSERT into Stock values (1180,20,800);
142 • INSERT into Stock values (1181,10,801);
143 • INSERT into Stock values (1182,10,802);
144
145 • INSERT into Discounts values (1200,20.00,'1month','active',800);
146 • INSERT into Discounts values (1201,50.00,'2month','expired',801);
147 • INSERT into Discounts values (1202,50.00,'1month','disabled',802);
148
149 • INSERT into Delivery values (400,'delivered','2023-08-03',200,100);
150 • INSERT into Delivery values (401,'delivered','2023-08-05',201,101);
151 • INSERT into Delivery values (402,'delivered','2023-08-02',202,102);
152
153 • INSERT into Income values ($00,'done',5000.00,200);
154 • INSERT into Income values ($01,'done',10000.00,201);
155 • INSERT into Income values ($02,'done',20000.00,202);
156
157 • INSERT into Employees values (600,'Jake','assistant',0765636120,200,100);
158 • INSERT into Employees values (601,'Cady','assistant',0765636100,201,101);
159 • INSERT into Employees values (602,'Jill','assistant',0765636200,202,102);
160

```

#	Time	Action	Message	Duration / Fetch
24	22:27:06	INSERT into Delivery values (402,'delivered','2023-08-02',202,102)	1 row(s) affected	0.015 sec
25	22:27:14	INSERT into Income values (\$00,'done',5000.00,200)	1 row(s) affected	0.000 sec
26	22:27:19	INSERT into Income values (\$01,'done',10000.00,201)	1 row(s) affected	0.000 sec
27	22:27:23	INSERT into Income values (\$02,'done',20000.00,202)	1 row(s) affected	0.000 sec
28	22:27:27	INSERT into Employees values (600,'Jake','assistant',0765636120,200,100)	1 row(s) affected	0.000 sec
70	22:27:31	INNOCOT into Employees values (601,'Cady','assistant',0765636100,201,101)	1 row(s) affected	0.000 sec

2. Use of querying tools. Perform following MySQL queries using the tables that have created in task 1.2. Provide evidence For MySQL queries and results.

2.1. Retrieve the details of a customer's order, including the clothing items ordered, their sizes, and the delivery status.

The screenshot shows the MySQL Workbench interface. The script editor contains a complex SELECT query joining multiple tables (Customers, Orders, Delivery, OrderedItems, ClothingItems) to retrieve details for a specific customer. The results are displayed in a 'Result Grid' pane, which shows a single row for customer ID 102. The bottom status bar indicates the session was completed at 11:07 PM on 8/10/2023.

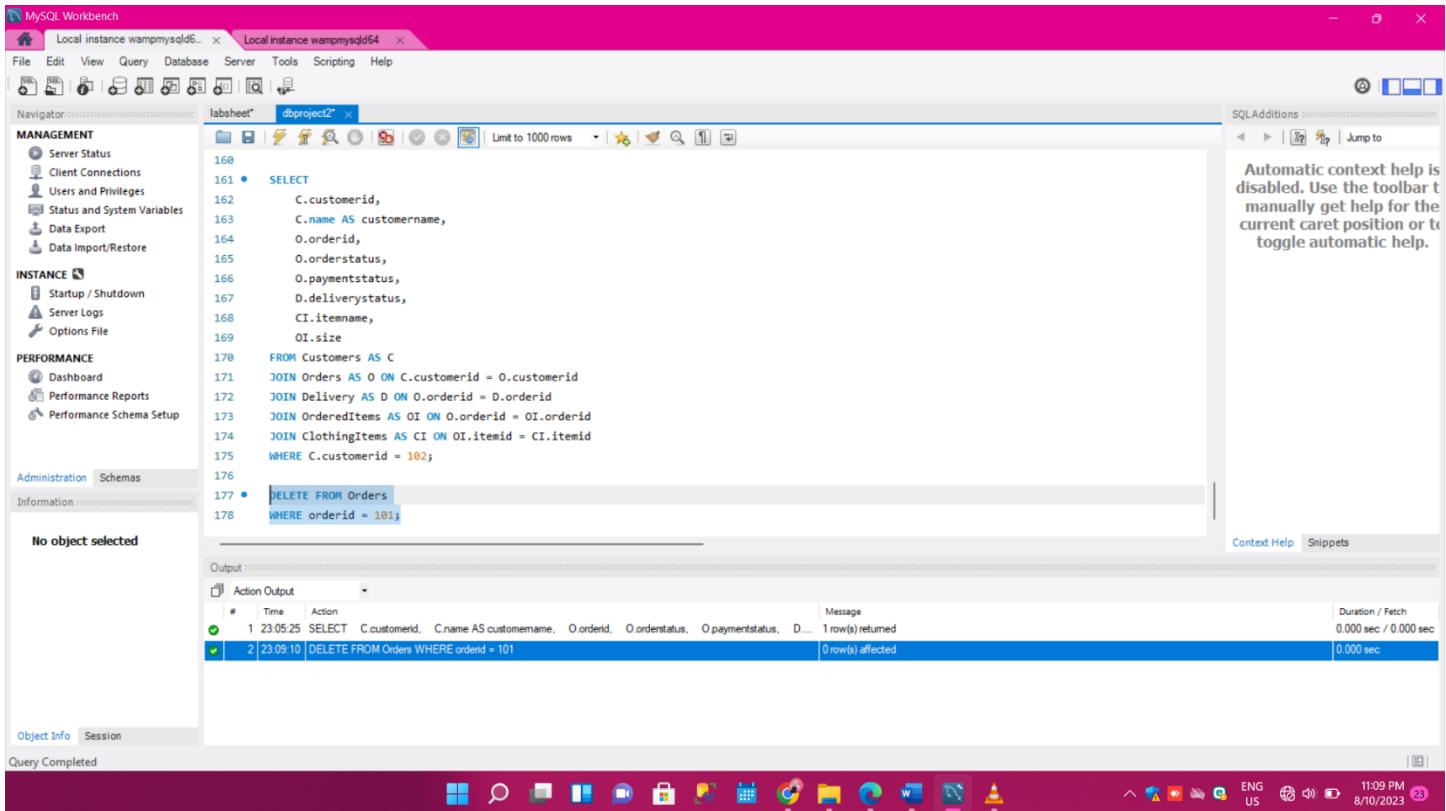
```

161 • SELECT
162     C.customerid,
163     C.name AS customername,
164     O.orderid,
165     O.orderstatus,
166     O.paymentstatus,
167     D.deliverystatus,
168     CI.itemname,
169     OI.size
170     FROM Customers AS C
171     JOIN Orders AS O ON C.customerid = O.customerid
172     JOIN Delivery AS D ON O.orderid = D.orderid
173     JOIN OrderedItems AS OI ON O.orderid = OI.orderid
174     JOIN ClothingItems AS CI ON OI.itemid = CI.itemid
175     WHERE C.customerid = 102;

```

customerid	customername	orderid	orderstatus	paymentstatus	deliverystatus	itemname	size
102	Naomi Perera	202	processing	fulfillment	delivered	highneck	I

2.2. Delete a specific customer's order along with all the associated order details, such as the clothing items and delivery information.



The screenshot shows the MySQL Workbench interface with a query editor window titled "labsheet" containing the following SQL code:

```

160
161 • SELECT
162     C.customerid,
163     C.name AS customername,
164     O.orderid,
165     O.orderstatus,
166     O.paymentstatus,
167     D.deliverystatus,
168     CI.itemname,
169     OI.size
170 FROM Customers AS C
171 JOIN Orders AS O ON C.customerid = O.customerid
172 JOIN Delivery AS D ON O.orderid = D.orderid
173 JOIN OrderdItems AS OI ON O.orderid = OI.orderid
174 JOIN ClothingItems AS CI ON OI.itemid = CI.itemid
175 WHERE C.customerid = 102;
176
177 • DELETE FROM Orders
178 WHERE orderid = 101;

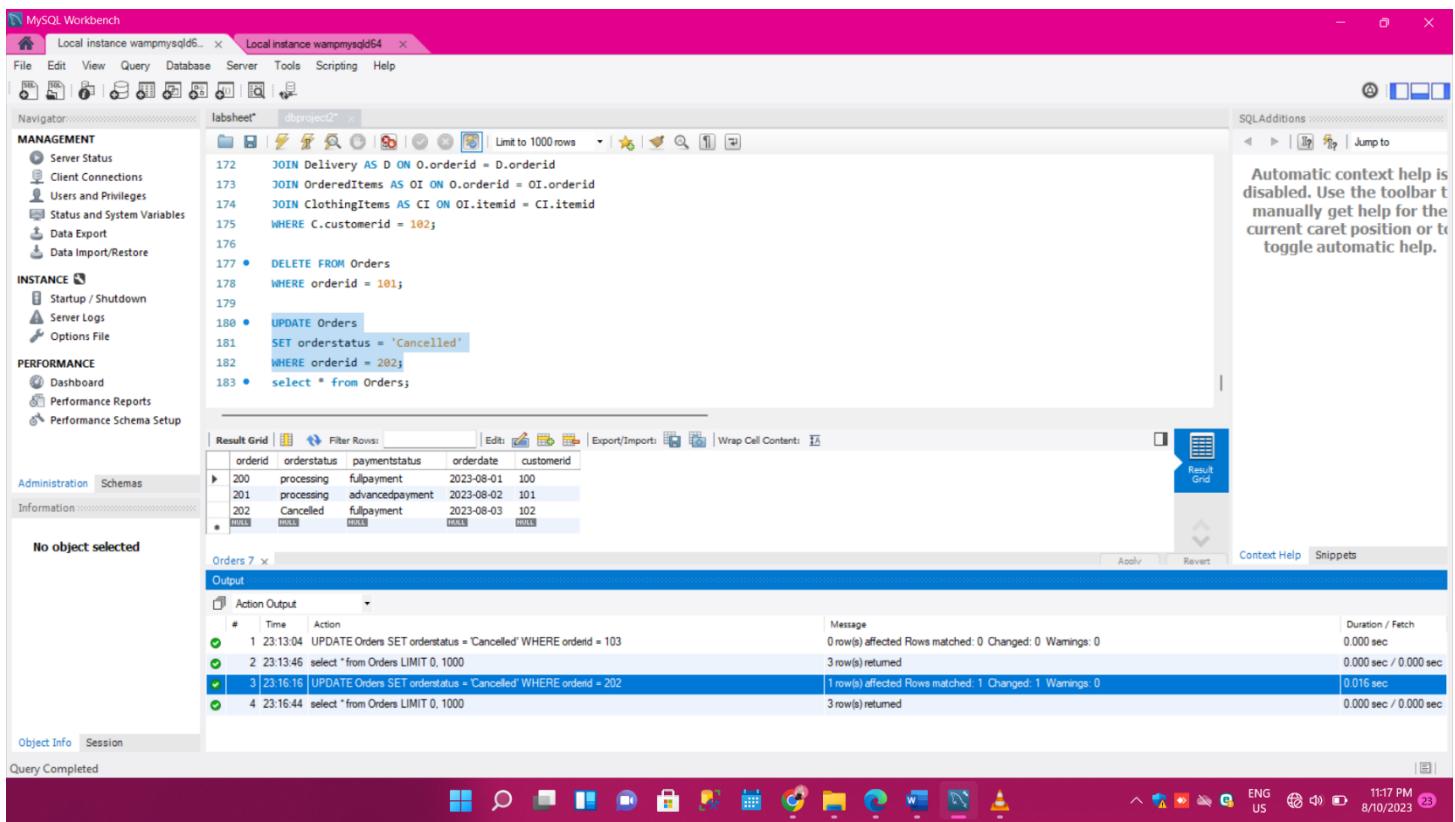
```

The "Output" pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	23:05:25	SELECT C.customerid, C.name AS customername, O.orderid, O.orderstatus, O.paymentstatus, D.deliverystatus, CI.itemname, OI.size FROM Customers AS C JOIN Orders AS O ON C.customerid = O.customerid JOIN Delivery AS D ON O.orderid = D.orderid JOIN OrderdItems AS OI ON O.orderid = OI.orderid JOIN ClothingItems AS CI ON OI.itemid = CI.itemid WHERE C.customerid = 102;	1 row(s) returned	0.000 sec / 0.000 sec
2	23:09:10	DELETE FROM Orders WHERE orderid = 101	0 row(s) affected	0.000 sec

The status bar at the bottom right indicates "11:09 PM 8/10/2023".

2.3. Update the customer's order status to "Cancelled" if they decide to cancel the order before delivery.



The screenshot shows the MySQL Workbench interface with a query editor window titled "labsheet" containing the following SQL code:

```

172 JOIN Delivery AS D ON O.orderid = D.orderid
173 JOIN OrderdItems AS OI ON O.orderid = OI.orderid
174 JOIN ClothingItems AS CI ON OI.itemid = CI.itemid
175 WHERE C.customerid = 102;
176
177 • DELETE FROM Orders
178 WHERE orderid = 101;
179
180 • UPDATE Orders
181 SET orderstatus = 'Cancelled'
182 WHERE orderid = 202;
183 • select * from Orders;

```

The "Result Grid" pane displays the updated order status:

orderid	orderstatus	paymentstatus	orderdate	customerid
200	processing	fulpayment	2023-08-01	100
201	processing	advancedpayment	2023-08-02	101
202	Cancelled	fulpayment	2023-08-03	102
• HOLE	HOLE	HOLE	HOLE	HOLE

The "Output" pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
1	23:13:04	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 103	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.000 sec
2	23:13:46	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
3	23:16:16	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 202	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
4	23:16:44	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

The status bar at the bottom right indicates "11:17 PM 8/10/2023".

2.4. Create a stored procedure to find the customer details and the order details who purchased any product from a given category.

The screenshot shows the MySQL Workbench interface with the 'dbproject2' database selected. In the central query editor, a stored procedure is being created:

```

176
177 • DELETE FROM Orders
178 WHERE orderid = 101;
179
180 • UPDATE Orders
181 SET orderstatus = 'Cancelled'
182 WHERE orderid = 202;
183 • select * from Orders;
184
185 DELIMITER //
186 • CREATE PROCEDURE GetAllOrdersByCategory(IN categoryname VARCHAR(255))
187 BEGIN
188     SELECT C.*, O.*
189     FROM Customers AS C
190     JOIN Orders AS O ON C.customerid = O.customerid
191     JOIN OrderedItems AS OI ON O.orderid = OI.orderid
192     JOIN ClothingItems AS CI ON OI.itemid = CI.itemid
193     WHERE CI.categoryid = (SELECT categoryid FROM ClothingCategory WHERE categoryname = categoryname);
194
195 END //
196 DELIMITER ;

```

The 'Output' pane shows the execution results:

Action	Time	Action	Message	Duration / Fetch
1	23:13:04	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 103	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.000 sec
2	23:13:46	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
3	23:16:16	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 202	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
4	23:16:44	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
5	23:21:48	CREATE PROCEDURE GetAllOrdersByCategory(IN categoryname VARCHAR(255)) BEGIN SELECT C.* O...	0 row(s) affected	0.016 sec

At the bottom right, the status bar shows '11:31 PM 8/10/2023'.

2.5 .Create a view to find the supplier details, product and the stock details as a summary.

The screenshot shows the MySQL Workbench interface with the 'dbproject2' database selected. In the central query editor, a view is being created:

```

186 • CREATE PROCEDURE GetAllOrdersByCategory(IN categoryname VARCHAR(255))
187 BEGIN
188     SELECT C.*, O.*
189     FROM Customers AS C
190     JOIN Orders AS O ON C.customerid = O.customerid
191     JOIN OrderedItems AS OI ON O.orderid = OI.orderid
192     JOIN ClothingItems AS CI ON OI.itemid = CI.itemid
193     WHERE CI.categoryid = (SELECT categoryid FROM ClothingCategory WHERE categoryname = categoryname);
194
195 END //
196 DELIMITER ;
197
198 • CREATE VIEW SupplierProductStockAllSummary AS
199     SELECT S.name AS name, CI.itemname, ST.quantity
200     FROM Suppliers AS S
201     JOIN SupplierItems AS SI ON S.supplierid = SI.supplierid
202     JOIN ClothingItems AS CI ON SI.itemid = CI.itemid
203     JOIN Stock AS ST ON CI.itemid = ST.itemid;
204

```

The 'Output' pane shows the execution results:

Action	Time	Action	Message	Duration / Fetch
1	23:13:04	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 103	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.000 sec
2	23:13:46	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
3	23:16:16	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 202	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
4	23:16:44	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
5	23:21:48	CREATE PROCEDURE GetAllOrdersByCategory(IN categoryname VARCHAR(255)) BEGIN SELECT C.* O...	0 row(s) affected	0.016 sec
6	23:30:34	CREATE VIEW SupplierProductStockAllSummary AS SELECT S.name AS name, CI.itemname, ST.quantity FR...	0 row(s) affected	0.000 sec

At the bottom right, the status bar shows '11:38 PM 8/10/2023'.

2.6. Create a trigger that automatically updates the stock quantity of clothing items according to the order payment status.

The screenshot shows the MySQL Workbench interface with a query editor window titled 'dbproject2'. The code in the editor is:

```

199 • CREATE VIEW SupplierProductStockAllSummary AS
200   SELECT S.name AS name, CI.itemname, ST.quantity
201   FROM Suppliers AS S
202   JOIN SupplierItems AS SI ON S.supplierid = SI.supplierid
203   JOIN ClothingItems AS CI ON SI.itemid = CI.itemid
204   JOIN Stock AS ST ON CI.itemid = ST.itemid;
205
206
207 • DELIMITER //
208 • CREATE TRIGGER UpdateStockQuantityOfClothingItemsOnPayment
209   AFTER UPDATE ON Orders FOR EACH ROW
210   BEGIN
211     IF NEW.paymentstatus = 'Full payment' THEN
212       UPDATE Stock
213         SET quantity = quantity - (SELECT SUM(quantity) FROM OrderedItems WHERE orderid = NEW.orderid)
214         WHERE itemid IN (SELECT itemid FROM OrderedItems WHERE orderid = NEW.orderid);
215     END IF;
216   END //
217   DELIMITER ;

```

The 'Output' tab shows the execution results:

#	Time	Action	Message	Duration / Fetch
2	23:13:46	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
3	23:16:16	UPDATE Orders SET orderstatus = 'Cancelled' WHERE orderid = 202	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec
4	23:16:44	select * from Orders LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
5	23:21:48	CREATE PROCEDURE GetAllOrdersByCategory(IN categoryname VARCHAR(255)) BEGIN SELECT C.* , O...	0 row(s) affected	0.016 sec
6	23:30:34	CREATE VIEW SupplierProductStockAllSummary AS SELECT S.name AS name, CI.itemname, ST.quantity FR...	0 row(s) affected	0.000 sec
7	23:36:57	CREATE TRIGGER UpdateStockQuantityOfClothingItemsOnPayment AFTER UPDATE ON Orders FOR EACH ...	0 row(s) affected	0.016 sec

2.7. Create two roles as ‘database admin’ and ‘stock keeper’ for the database. Create users under ‘database admin’ and ‘stock keeper’. Grant following privileges to users under each role.

- Database admin - Grant full access to all tables, allowing users assigned to this role to access and modify any table within the database.
- Stock keeper - Grant CRUD (Create, Read, Update, Delete) operations only for the stocks, ensuring that users in this role can only perform actions related to managing stock data.

MySQL Workbench

Local instance wampmysqld6... Local instance wampmysqld6...

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet* dbproject2*

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas Information

No object selected

Output:

#	Time	Action	Message	Duration / Fetch
1	23:47:05	CREATE ROLE databaseadmin	0 row(s) affected	0.015 sec
2	23:47:12	CREATE ROLE stockkeeper	0 row(s) affected	0.000 sec
3	23:53:29	CREATE user 'adminuser'@'localhost' identified by '1234';	0 row(s) affected	0.000 sec
4	23:53:35	CREATE user 'stockuser'@'localhost' identified by '1111';	0 row(s) affected	0.016 sec

Object Info Session

Query Completed

11:55 PM 8/10/2023

MySQL Workbench

Local instance wampmysqld6... Local instance wampmysqld6...

File Edit View Query Database Server Tools Scripting Help

Navigator: labsheet* dbproject2*

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas Information

No object selected

Output:

#	Time	Action	Message	Duration / Fetch
1	00:00:30	GRANT ALL privileges on czonlineclothingstore.* to 'adminuser'@'localhost'	0 row(s) affected	0.000 sec
2	00:03:54	GRANT select, insert, update, delete on czonlineclothingstore.Stock to 'stockuser'@'localhost'	0 row(s) affected	0.000 sec
3	00:05:27	GRANT databaseadmin TO adminuser	Error Code: 3523. Unknown authorization ID 'adminuser'@'%'	0.000 sec
4	00:06:03	GRANT databaseadmin TO 'adminuser'@'localhost'	0 row(s) affected	0.000 sec
5	00:07:14	GRANT stockkeeper TO 'stockuser'@'localhost'	0 row(s) affected	0.000 sec

Object Info Session

Query Completed

12:07 AM 8/11/2023

Database admin can do anything like deleting a table, adding a table or modifying a table.

MySQL Workbench

Local instance wampmysqld4 databaseadmin

File Edit View Query Database Server Tools Scripting Help

Navigator

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

No object selected

Object Info Session

Query Completed

Output:

#	Time	Action	Message	Duration / Fetch
1	14:55:21	set role databaseadmin	0 row(s) affected	0.000 sec
2	14:55:26	use czonlineclothingstore	0 row(s) affected	0.000 sec
3	14:55:29	update ClothingCategory set categoryname='skirt' where categoryid=701;	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.016 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

2.55 PM 8/23/2023

For the stock keeper he is only able to modify stock data, so if you try to modify something else , you should get an error.

MySQL Workbench

Local instance wampmysqld4 databaseadmin stockkeeper

File Edit View Query Database Server Tools Scripting Help

Navigator

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

No object selected

Object Info Session

Output:

#	Time	Action	Message	Duration / Fetch
1	14:58:02	set role stockkeeper	0 row(s) affected	0.000 sec
2	14:58:06	use czonlineclothingstore	0 row(s) affected	0.000 sec
3	14:58:09	delete from ClothingCategory where categoryid=702;	Error Code: 1142. DELETE command denied to user 'stockuser'@'localhost' for table 'clothingcategory'	0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Context Help Snippets

2.59 PM 8/23/2023

REFERENCES

- ➡ <https://app.diagrams.net/>- Design the ER diagram
- ➡ Used MySQL Workbench in order to design the database.
- ➡ www.w3schools.com – References of MySQL, Query tools
- ➡ YouTube videos on Granting Privileges.
- ➡ Used the PowerPoint presentations pdf on ER Diagrams & Relational Mapping, View, Procedure, Triggers and Database Security (owner Ms.Hansika Samanthi)
- ➡ Used the PowerPoint presentation pdf on SQL (owner Ms.Mahesha Thejani)