

DDL CONSTRAINTS OUTPUTS

Query:

```
create database Sales;
use Sales;
create table Orders(
  OrderId int auto_increment primary key,
  CustomerName varchar(30) not null,
  ProductCategory varchar(30) not null,
  Ordered_item varchar(30) not null,
  ContactNo int unique
);
desc Orders;
```

The screenshot displays the MySQL Workbench interface. The SQL Editor window contains the following queries:

```
4 OrderId int auto_increment primary key,
5 CustomerName varchar(30) not null,
6 ProductCategory varchar(30) not null,
7 Ordered_item varchar(30) not null,
8 ContactNo int unique
9 );
10 desc Orders;
```

The Result Grid shows the output of the `desc Orders;` query:

Field	Type	Null	Key	Default	Extra
OrderId	int	NO	PRI	auto_increment	
CustomerName	varchar(30)	NO			
ProductCategory	varchar(30)	NO			
Ordered_item	varchar(30)	NO			
ContactNo	int	YES	UNI		

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
22	13:35:33	create database Sales	1 row(s) affected	0.015 sec
23	13:35:39	use Sales	0 row(s) affected	0.016 sec
24	13:38:42	create table Orders(OrderId int auto_increment primary key, CustomerName varc...	0 row(s) affected	0.047 sec
25	13:38:56	desc Orders	5 row(s) returned	0.000 sec / 0.000 sec

1. Add a new column named "order_quantity" to the orders table.

Query:

```
ALTER TABLE Orders add column Order_quantity int;  
desc Orders;
```

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

```
6 ProductCategory varchar(30) not null,  
7 Ordered_item varchar(30) not null,  
8 ContactNo int unique  
9 );  
10 desc Orders;  
11  
12 ALTER TABLE Orders add column Order_quantity int;
```

The left sidebar shows the database schema with the 'orders' table selected. The 'Table: orders' information panel shows the following columns:

Columns:	OrderId	CustomerName	ProductCategory	Ordered_item	ContactNo
	int AI P	varchar	varchar	varchar	int

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

Field	Type	Null	Key	Default	Extra
CustomerName	varchar(30)	NO			
ProductCategory	varchar(30)	NO			
Ordered_item	varchar(30)	NO			
ContactNo	int	YES	UNI		
Order_quantity	int	YES			

The 'Output' panel shows the execution log:

#	Time	Action	Message	Duration / Fetch
24	13:38:42	create table Orders(OrderId int auto_increment primary key, CustomerName var...	0 row(s) affected	0.047 sec
25	13:38:56	desc Orders	5 row(s) returned	0.000 sec / 0.000 sec
26	13:42:37	ALTER TABLE Orders add column Order_quantity int	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec
27	13:42:42	desc Orders	6 row(s) returned	0.000 sec / 0.000 sec

2. Rename the orders table to the sales_orders table.

Query:

```
rename table Orders to SalesOrders;  
desc SalesOrders;
```

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

```
9 );  
10 desc Orders;  
11  
12 ALTER TABLE Orders add column Order_quantity int;  
13  
14 rename table Orders to SalesOrders;  
15 desc SalesOrders;
```

The left sidebar shows the database schema with the 'salesorders' table selected. The 'Table: salesorders' information panel shows the following columns:

Columns:	OrderId	CustomerName	ProductCategory	Ordered_item	ContactNo	Order_quantity
	int AI P	varchar	varchar	varchar	int	int

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

Field	Type	Null	Key	Default	Extra
OrderId	int	NO	PRI		auto_increment
CustomerName	varchar(30)	NO			
ProductCategory	varchar(30)	NO			
Ordered_item	varchar(30)	NO			
ContactNo	int	YES	UNI		
Order_quantity	int	YES			

The 'Output' panel shows the execution log:

#	Time	Action	Message	Duration / Fetch
28	13:45:37	rename table Orders to SalesOrders	0 row(s) affected	0.031 sec
29	13:45:55	desc SalesOrders	6 row(s) returned	0.000 sec / 0.000 sec
30	13:46:28	Select * from salesorders LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
31	13:46:38	desc SalesOrders	6 row(s) returned	0.016 sec / 0.000 sec

3. Insert 10 rows into the sales_orders table.

Query:

```
INSERT INTO SalesOrders (CustomerName, ProductCategory, Ordered_item, ContactNo,
Order_quantity) VALUES
('John Doe', 'Electronics', 'Smartphone', '1234567890', 2),
('Jane Smith', 'Clothing', 'Jacket', '0987654321', 1),
('Alice Johnson', 'Home Appliances', 'Microwave', '1122334455', 3),
('Bob Brown', 'Furniture', 'Sofa', '5566778899', 1),
('Chris Green', 'Books', 'Novel', '6677889900', 5),
('Eve White', 'Groceries', 'Apples', '7788990011', 10),
('David Black', 'Toys', 'Action Figure', '8899001122', 2),
('Sophia Gray', 'Sports', 'Tennis Racket', '9900112233', 1),
('Michael Blue', 'Automotive', 'Car Tires', '1100223344', 4),
('Olivia Purple', 'Jewelry', 'Necklace', '2211334455', 1);
select * from salesorders;
```

The screenshot shows the MySQL Workbench interface. The SQL Editor at the top contains the INSERT query and a SELECT statement to verify the data. The Results Grid at the bottom displays the data inserted into the salesorders table.

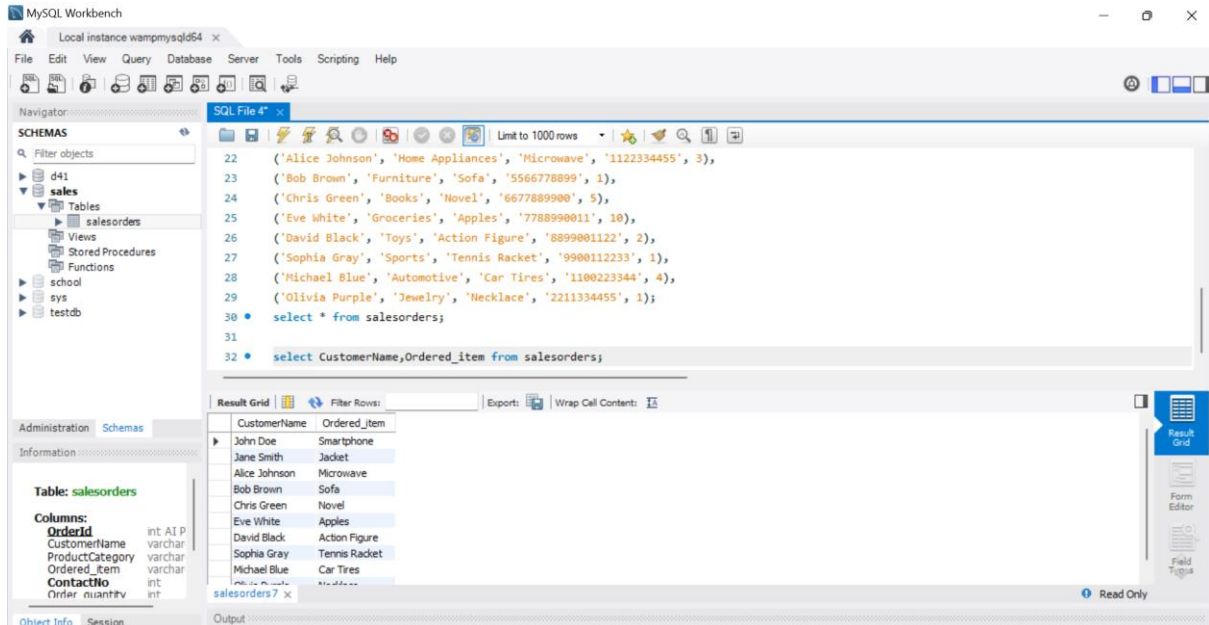
Table: salesorders

OrderId	CustomerName	ProductCategory	Ordered_item	ContactNo	Order_quantity
21	John Doe	Electronics	Smartphone	1234567890	2
22	Jane Smith	Clothing	Jacket	0987654321	1
23	Alice Johnson	Home Appliances	Microwave	1122334455	3
24	Bob Brown	Furniture	Sofa	5566778899	1
25	Chris Green	Books	Novel	6677889900	5
26	Eve White	Groceries	Apples	7788990011	10
27	David Black	Toys	Action Figure	8899001122	2
28	Sophia Gray	Sports	Tennis Racket	9900112233	1
29	Michael Blue	Automotive	Car Tires	1100223344	4
30	Olivia Purple	Jewelry	Necklace	2211334455	1

4. Retrieve customer_name and Ordered_Item from the sales_orders table

Query:

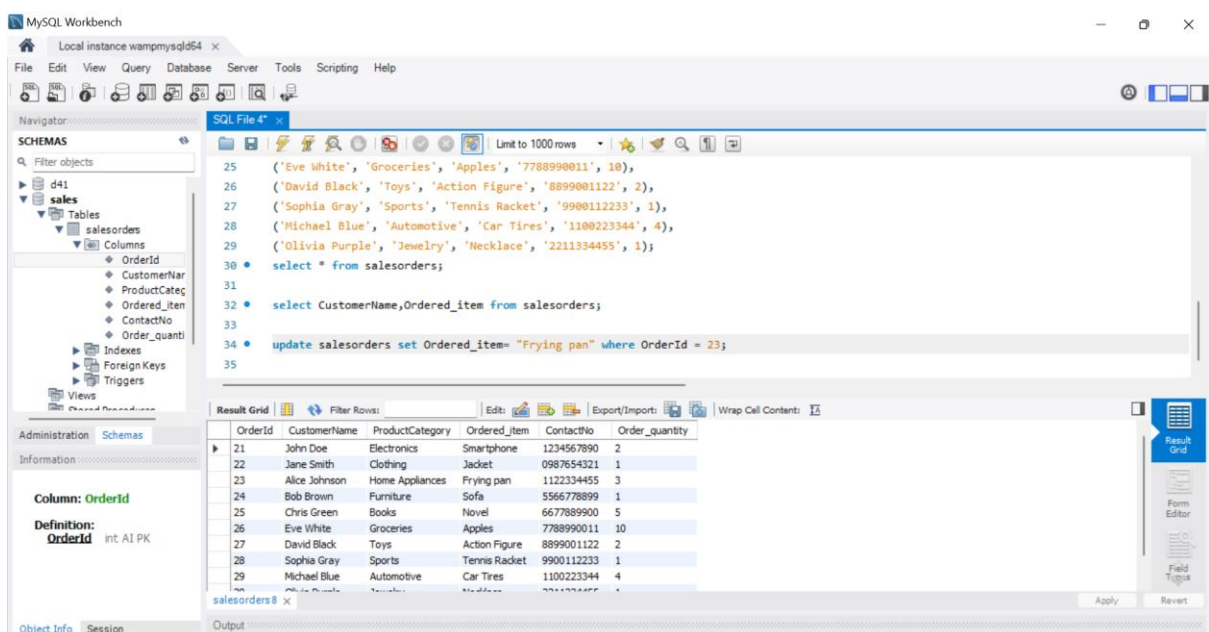
```
select CustomerName,Ordered_item from salesorders;
```



5. Use the update command to change the name of the product for any row

Query:

```
update salesorders set Ordered_item= "Frying pan" where OrderId = 23;  
select * from salesorders;
```



6. Delete the sales_orders table from the database.

Query:

```
drop table salesorders;  
show tables;
```

