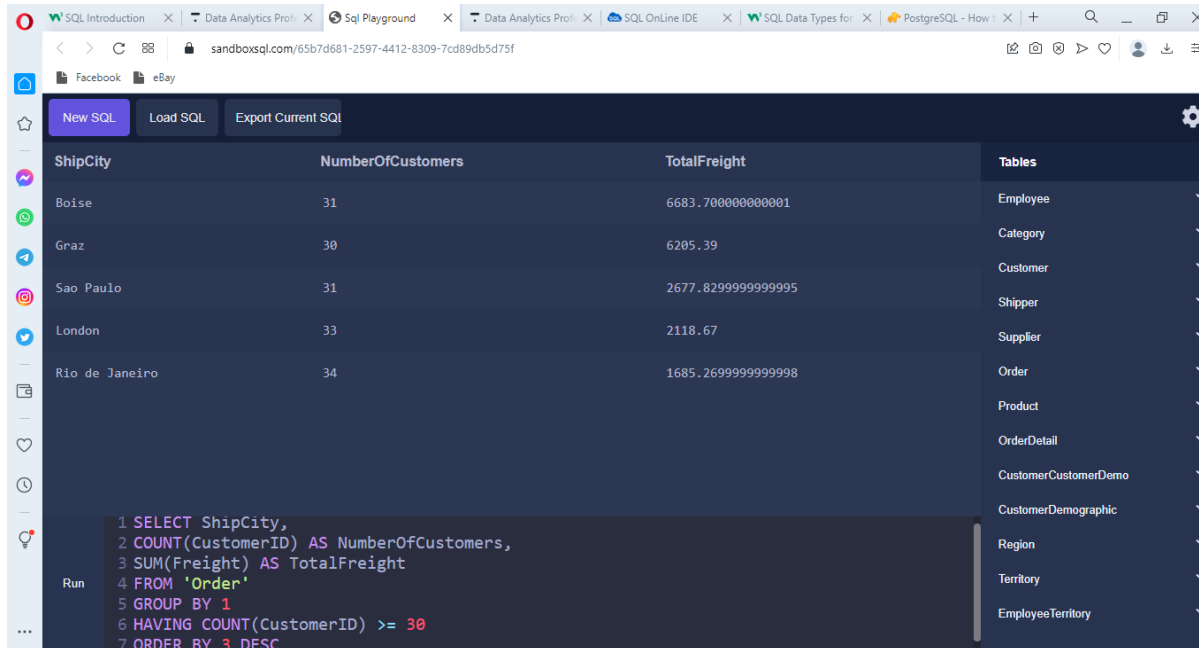


Shipment Analysis Using SQL

This project was carried out by Chidiebere David Ogbonna.

1. What are the total sums of freights of ship cities that have 30 or more customers?



The screenshot shows a web browser with multiple tabs, including 'SQL Playground'. The URL is 'sandboxsql.com/65b7d681-2597-4412-8309-7cd89db5d75f'. The interface has a dark theme. At the top, there are buttons for 'New SQL', 'Load SQL', and 'Export Current SQL'. Below these is a table with the following data:

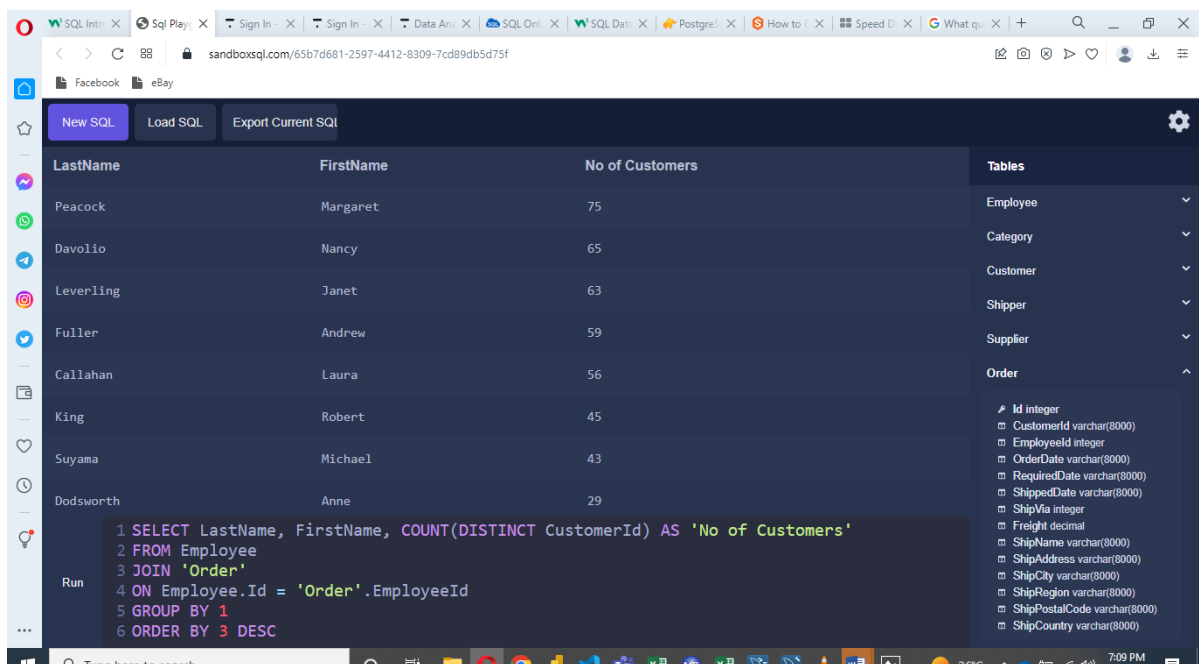
ShipCity	NumberOfCustomers	TotalFreight
Boise	31	6683.700000000001
Graz	30	6205.39
Sao Paulo	31	2677.8299999999995
London	33	2118.67
Rio de Janeiro	34	1685.2699999999998

On the right side, there is a 'Tables' list with the following items: Employee, Category, Customer, Shipper, Supplier, Order, Product, OrderDetail, CustomerCustomerDemo, CustomerDemographic, Region, Territory, and EmployeeTerritory. At the bottom, there is a SQL query editor with the following code:

```
1 SELECT ShipCity,
2 COUNT(CustomerID) AS NumberOfCustomers,
3 SUM(Freight) AS TotalFreight
4 FROM 'Order'
5 GROUP BY 1
6 HAVING COUNT(CustomerID) >= 30
7 ORDER BY 3 DESC
```

The 'Run' button is visible next to the query editor.

2. Generate a table of employees and the number of customers they handle.



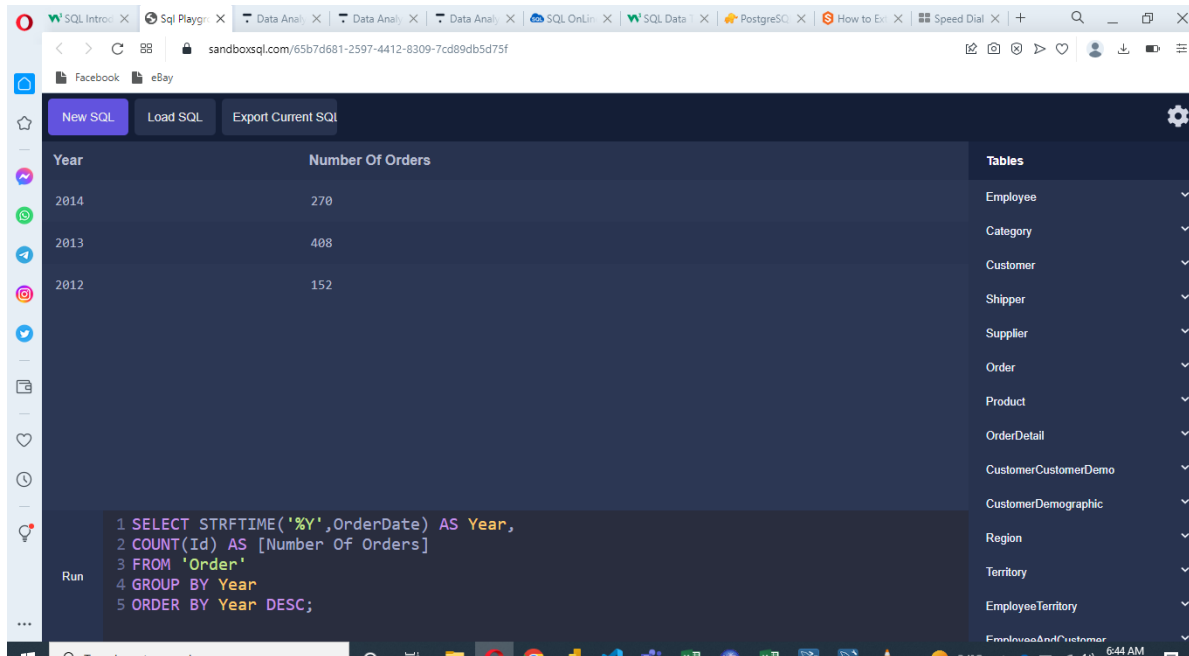
The screenshot shows the same SQL Playground interface. The query editor now contains the following code:

```
1 SELECT LastName, FirstName, COUNT(DISTINCT CustomerId) AS 'No of Customers'
2 FROM Employee
3 JOIN 'Order'
4 ON Employee.Id = 'Order'.EmployeeId
5 GROUP BY 1
6 ORDER BY 3 DESC
```

The 'Run' button is visible next to the query editor. The results table is not yet displayed, but the 'Tables' list on the right is expanded, showing the 'Order' table structure:

- Id integer
- CustomerId varchar(8000)
- EmployeeId integer
- OrderDate varchar(8000)
- RequiredDate varchar(8000)
- ShippedDate varchar(8000)
- ShipVia integer
- Freight decimal
- ShipName varchar(8000)
- ShipAddress varchar(8000)
- ShipCity varchar(8000)
- ShipRegion varchar(8000)
- ShipPostalCode varchar(8000)
- ShipCountry varchar(8000)

3. How many orders were made each year?



The screenshot shows a web-based SQL IDE interface. The main window displays a table with two columns: 'Year' and 'Number Of Orders'. The data is as follows:

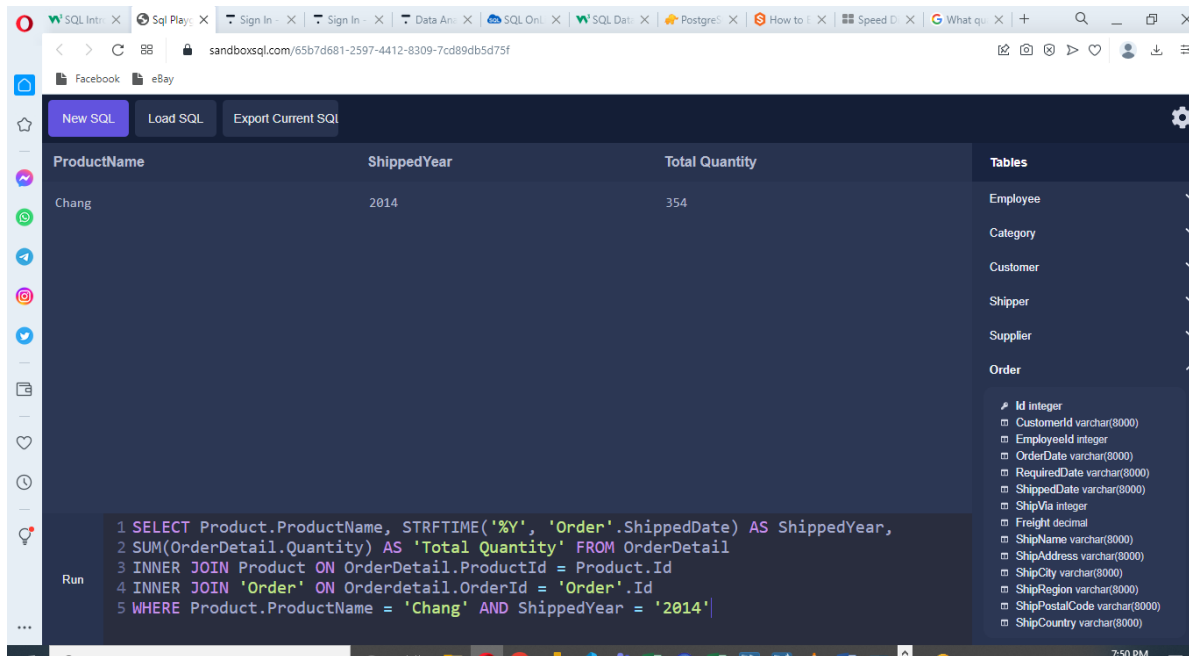
Year	Number Of Orders
2014	270
2013	408
2012	152

Below the table, the SQL query is visible:

```
1 SELECT STRFTIME('%Y',OrderDate) AS Year,  
2 COUNT(Id) AS [Number Of Orders]  
3 FROM 'Order'  
4 GROUP BY Year  
5 ORDER BY Year DESC;
```

The right sidebar shows a list of tables: Employee, Category, Customer, Shipper, Supplier, Order, Product, OrderDetail, CustomerCustomerDemo, CustomerDemographic, Region, Territory, EmployeeTerritory, and EmployeeAndCustomer.

4. What quantity of the product 'Chang' was shipped in 2014?



The screenshot shows the same web-based SQL IDE interface. The main window displays a table with three columns: 'ProductName', 'ShippedYear', and 'Total Quantity'. The data is as follows:

ProductName	ShippedYear	Total Quantity
Chang	2014	354

Below the table, the SQL query is visible:

```
1 SELECT Product.ProductName, STRFTIME('%Y', 'Order'.ShippedDate) AS ShippedYear,  
2 SUM(OrderDetail.Quantity) AS 'Total Quantity' FROM OrderDetail  
3 INNER JOIN Product ON OrderDetail.ProductId = Product.Id  
4 INNER JOIN 'Order' ON OrderDetail.OrderId = 'Order'.Id  
5 WHERE Product.ProductName = 'Chang' AND ShippedYear = '2014';
```

The right sidebar shows a list of tables, with the 'Order' table expanded to show its fields: Id integer, CustomerId varchar(8000), EmployeeId integer, OrderDate varchar(8000), RequiredDate varchar(8000), ShippedDate varchar(8000), ShipVia integer, Freight decimal, ShipName varchar(8000), ShipAddress varchar(8000), ShipCity varchar(8000), ShipRegion varchar(8000), ShipPostalCode varchar(8000), and ShipCountry varchar(8000).

5. What are the revenues per supplier in 2013 before and after discount applied? Show both values in a single table.

CompanyName	ShippedYear	Revenue Before Discount	Revenue After Discount	Tables
Plutzer Lebensmittelgroßmärkte AG	2013	65170.02000000002	62187.286000000015	Employee
Gai pâturage	2013	57674.2	54268.83	Category
Pavlova, Ltd.	2013	57176.74999999999	52438.6775	Customer
Aux joyeux ecclésiastiques	2013	54255.600000000006	51038.784999999996	Shipper
G'day, Mate	2013	39919.4	38402.299999999996	Supplier
Pasta Buttini s.r.l.	2013	37036.3	34760.7	Order
Forêts d'érables	2013	32275.8	29854.325000000008	Product
Norske Meierier	2013	27015.199999999997	24470.549999999996	OrderDetail

```
1 SELECT Supplier.CompanyName,
2 STRFTIME('%Y', 'Order'.ShippedDate) AS ShippedYear,
3 SUM(OD.unitprice * OD.Quantity) AS 'Revenue Before Discount',
4 SUM((OD.unitprice * OD.Quantity) - (OD.unitprice * OD.Quantity * OD.Discount))
5 AS 'Revenue After Discount'
6 FROM OrderDetail AS OD
```

CompanyName	ShippedYear	Revenue Before Discount	Revenue After Discount	Tables
Plutzer Lebensmittelgroßmärkte AG	2013	65170.02000000002	62187.286000000015	Employee
Gai pâturage	2013	57674.2	54268.83	Category
Pavlova, Ltd.	2013	57176.74999999999	52438.6775	Customer
Aux joyeux ecclésiastiques	2013	54255.600000000006	51038.784999999996	Shipper
G'day, Mate	2013	39919.4	38402.299999999996	Supplier
Pasta Buttini s.r.l.	2013	37036.3	34760.7	Order
Forêts d'érables	2013	32275.8	29854.325000000008	Product
Norske Meierier	2013	27015.199999999997	24470.549999999996	OrderDetail

```
7 LEFT JOIN 'Order' ON OD.OrderId = 'Order'.Id
8 LEFT JOIN Product ON OD.ProductId = Product.Id
9 LEFT JOIN Supplier ON Product.SupplierId = Supplier.Id
10 WHERE ShippedYear = '2013'
11 GROUP BY 1
12 ORDER BY 3 DESC;
```