Award Sales-incentive Bonuses Using the W3Schools SQL Tool

Toni Randell

The goal of this project was to find out which employees had the five highest order amounts so that the company could issue them a raise.

Seeked solution:

A list of employees who have orders with the five highest sales amounts.

This project was ran using the W3Schools SQL editor. There are 8 pre uploaded tables entitled the following: Customers, Categories, Employees, OrderDetails, Orders, Products, Shippers, Suppliers.

Now, with the goal in mind, I began to write my queries.

There were 10 total employees. I found this by using the following query:

SELECT count(*) FROM employees;

This table did not include any sales information, only information about the employees themselves.

Next, I ran the following query:

SELECT count(*) FROM orders;

I was interested in seeing the scope that we were working with. I do not think that this information is vital to completing the goal, however, it allowed me to know what I was working with. There was a total of 196 orders.

Next, I ran the following query to see what information (data) was included in the table:

Award Sales-incentive Bonuses Using the W3Schools SQL Tool

Toni Randell

Select * From orders;

This table did not give me much information in regards to the actual revenue per oder, it listed the, order id, customer id, employee id, order date, and the shipper id.

Next, I utilized the following query to zoom into more information regarding the orders:

Select * FROM OrderDetails

ORDER BY OrderID

Next I used the following query:

SELECT lastname, firstname, orderid

From Employees

INNER JOIN Orders ON Employees.EmployeeID=Orders.EmployeeID

ORDER BY lastname,firstname

This join allowed me to visualize the order identifier with the employee that was responsible for that order.

Next I tried to use the following query but it gave me an error:

SELECT lastname, firstname, orders.orderid, products.productid, quantity, price

FROM employees

INNER JOIN orders ON employees.employeeID = orders.employeeid

INNER JOIN orderdetails ON orders.orderid = orderdetails.orderid

INNER JOIN products ON orderdetails.productid = products.productid

ORDER BY lastname, firstname;

Award Sales-incentive Bonuses Using the W3Schools SQL Tool

Toni Randell

Doing all of this still did not work. I asked ChatGPT if there was something off logically about my query and it said there wasn't, which I didn't think there was. My next thought was to email W3 schools to ask them about the query, because I figured that their editor must just be requiring that it be written a certain way. They instructed me to try the following:

```
SELECT lastname, firstname, orders.orderid, products.productid, quantity, price
FROM (((employees
  INNER JOIN orders ON employees.employeeID = orders.employeeid)
  INNER JOIN orderdetails ON orders.orderid = orderdetails.orderid)
  INNER JOIN products ON orderdetails.productid = products.productid)
ORDER BY lastname, firstname;
```

And this query worked. Yay, trouble shooting! The purpose for this query was to present all of the information needed to answer the question in one table view.

Next I used the following query to find the total sales amount for each order:

```
SELECT lastname, firstname, orders.orderid, sum(quantity * price) as SalesAmt
FROM (((employees
  INNER JOIN orders ON employees.employeeID = orders.employeeid)
  INNER JOIN orderdetails ON orders.orderid = orderdetails.orderid)
  INNER JOIN products ON orderdetails.productid = products.productid)
Group by orders.orderid, lastname, firstname
```

I also initially had an issue with the query I tried before getting this one, so there was some trouble shooting involved here as well. I realized that I did not need so select a few of my columns.

Next I used the following query to list the employees with the highest sales amounts:

Award Sales-incentive Bonuses Using the W3Schools SQL Tool

Toni Randell

SELECT TOP 5 lastname, firstname, orders.orderid, sum(quantity * price) as SalesAmt

FROM (((employees

  INNER JOIN orders ON employees.employeeID = orders.employeeid)

  INNER JOIN orderdetails ON orders.orderid = orderdetails.orderid)

  INNER JOIN products ON orderdetails.productid = products.productid)

Group by orders.orderid, lastname, firstname

Order by sum(quantity*price) desc

Number of Records: 5

lastname	firstname	orderid	SalesAmt
Buchanan	Steven	10372	15353.6
King	Robert	10424	14366.5
Peacock	Margaret	10417	14104
King	Robert	10353	13427
Peacock	Margaret	10360	9244.25

As you can see, this query only includes 3 people. Because two people both had 2 of the top orders. Therefore, I had to look at the joined, and calculated data again to find who actually received the top orders. Then I used the order ids to query the 5 people with the highest sales amount. See below:

SELECT TOP 5 lastname, firstname, orders.orderid, sum(quantity * price) as SalesAmt

FROM (((employees

  INNER JOIN orders ON employees.employeeID = orders.employeeid)

  INNER JOIN orderdetails ON orders.orderid = orderdetails.orderid)

  INNER JOIN products ON orderdetails.productid = products.productid)

Group by orders.orderid, lastname, firstname

Award Sales-incentive Bonuses Using the W3Schools SQL Tool

Toni Randell

having orders.orderid in (10372, 10424, 10417, 10324, 10351)

Order by sum(quantity*price) desc

Number of Records: 5

lastname	firstname	orderid	SalesAmt
Buchanan	Steven	10372	15353.6
King	Robert	10424	14366.5
Peacock	Margaret	10417	14104
Dodsworth	Anne	10324	7698.45
Davolio	Nancy	10351	7103.6

Success! We have answered the question presented to us by the fictional business!

Throughout the project, I continued to refer to what the goal was issued by the fictional company so that in my Data Exploration, I did not get misguided to the wrong paths and started making connections that, while they may be useful in other contexts, were not relevant to the goal at hand.

Also, along the way, I had some troubleshooting that occurred that took me a day to figure out, waiting for the email response. This was somewhat frustrating because I find SQL to be very easy and could not understand where the logical error was in my query. I actually had to email W3Schools support 3 times because the queries I was attempting to run were not compatible with their editor. I could have given up on this project and chosen another one because how would anyone have known technically, but I decided to push through. 🙂