CAP2761-Asssignments / Assignment_12.ipynb

abdulrab2101 Assignment Complete fbee46e · now

690 lines (690 loc) · 25.8 KB

Preview Code Blame Raw

7.1 Compute an Average

Question 1: Use the table `Sales.SalesOrderHeader` to find the average freight per year.

In [1]:
```
SELECT YEAR(OrderDate) AS OrderYear, AVG(Freight) AS AverageFreight
FROM Sales.SalesOrderHeader
GROUP BY YEAR(OrderDate);
```

(4 rows affected)
Total execution time: 00:00:00.464

Out[1]:

OrderYear	AverageFreight
2013	89.7081
2014	47.8087
2011	224.3203
2012	252.4721

Question 2: Order by year the result of Question 1.

In [2]:
```
SELECT YEAR(OrderDate) AS OrderYear, AVG(Freight) AS AverageFreight
FROM Sales.SalesOrderHeader
GROUP BY YEAR(OrderDate)
ORDER BY OrderYear;
```

(4 rows affected)
Total execution time: 00:00:00.050

Out[2]:

OrderYear	AverageFreight
2011	224.3203
2012	252.4721
2013	89.7081
2014	47.8087

Question 3: Explanation - Is the average freight generally increasing or decreasing?

Explaination:

We can see that the Average frieght is decreasing every year except 2012 where It increased a little.

7.2 Count Rows in a Group

Question 4: Explanation - What is the difference between `COUNT` and `COUNT_BIG` ?

Explanation:

- `COUNT` : Returns the number of items in a group. It's suitable for most applications where the result count is within the range of the `int` data type.
- `COUNT_BIG` : Returns the number of items in a group as a `bigint` data type, which can handle much larger results than `COUNT` .

7.3 Sum Values in a Group

Question 5: Use the `SUM` function to add up the `TaxAmt` of the `Sales.SalesOrderHeader` .

In [3]:
```
SELECT SUM(TaxAmt) AS TotalTaxAmount
FROM Sales.SalesOrderHeader;
```

(1 row affected)

Total execution time: 00:00:00.071

Out[3]:

TotalTaxAmount
10186974.4602

Question 6: Return only results for the year 2012 grouped and ordered by month.

In [4]:
```
SELECT MONTH(OrderDate) AS OrderMonth, SUM(TaxAmt) AS TotalTaxAmount
FROM Sales.SalesOrderHeader
WHERE YEAR(OrderDate) = 2012
GROUP BY MONTH(OrderDate)
ORDER BY OrderMonth;
```

(12 rows affected)

Total execution time: 00:00:00.083

Out[4]:

OrderMonth	TotalTaxAmount
1	371588.69
2	132285.70
3	274742.2648
4	180817.2923
5	288245.0511
6	389556.4568
7	321735.2997
8	203286.8258
9	325769.325
10	239214.5416
11	171010.3965
12	264718.7376

7.4 Find High and Low Values in a Group

Question 7: Find the `Min TaxRate` from the table `Sales.SalesTaxRate` .

In [5]:
```
SELECT MIN(TaxRate) AS MinTaxRate
FROM Sales.SalesTaxRate;
```

(1 row affected)

Total execution time: 00:00:00.051

Out[5]:

MinTaxRate

5.00

Question 8: Find the `Max TaxRate` from the table `Sales.SalesTaxRate` .

In [6]:
```
SELECT MAX(TaxRate) AS MaxTaxRate
FROM Sales.SalesTaxRate;
```

(1 row affected)

Total execution time: 00:00:00.021

Out[6]:

MaxTaxRate
19.60

7.6 Find Variance of Values in a Column

Question 9: Find the `Variance` of the `TaxRate` from the table `Sales.SalesTaxRate` .

In [7]:
```
SELECT VAR(TaxRate) AS VarianceTaxRate
FROM Sales.SalesTaxRate;
```

(1 row affected)

Total execution time: 00:00:00.056

Out[7]:

VarianceTaxRate
14.235548029556671

7.7 Find Std Dev of Values in a Column

Question 10: Find the `StandardDeviation` of the `TaxRate` from the table `Sales.SalesTaxRate` .

In [8]:
```
SELECT STDEV(TaxRate) AS StdDevTaxRate
FROM Sales.SalesTaxRate;
```

(1 row affected)

Total execution time: 00:00:00.076

Out[8]:

StdDevTaxRate
3.7730025218062964