1.1 Use the information\_schema to find out how many rows there are in each table in the

adventureworks data warehouse. Show the table name and its row count.

Hints:

• use information\_schema;

• There is a table within information\_schema called TABLES.

**-- Q1**

**use information\_schema;**

**Select table\_name, table\_rows**

**from tables**

**where table\_schema like 'aw%';**

**Table

Description automatically generated**

1.2 Then run a SELECT COUNT(\*) against each table and compare the row count

from the SELECT COUNT(\*) to the result from the information\_schema.

|  |  |
| --- | --- |
| DimAccount | 99 |
| DimCurrency | 0 |
| DimCustomer | 18483 |
| DimDepartmentGroup | 7 |
| DimEmployee | 296 |
| DimGeography | 655 |
| DimOrganization | 14 |
| DimProduct | 158 |
| DimProductCategory | 4 |
| DimProductSubcategory | 37 |
| DimPromotion | 16 |
| DimReseller | 701 |
| DimSalesReason | 10 |
| DimSalesTerritory | 11 |
| DimScenario | 3 |
| DimTime | 1158 |
| FactCurrencyRate | 0 |
| FactFinance | 39410 |
| FactInternetSales | 60398 |

1.3 Do some research to find out why MySQL might return different values in such a

scenario. Document your findings. In other words, explain WHY is the row count

from the information\_schema might sometimes be different from the SELECT

COUNT(\*).

**Based on my research, information\_schema usually estimate the count of rows, so the value we saw with information\_schema is an estimated value. On the other hands, information\_schema may only count NON\_Null value while SELECT COUNT(\*) doesn’t. The row count of information\_schema may be calculated based on the statistical sampling techniques to estimate the number.**

1.4 Why is the select against the information\_schema more efficient (although perhaps

less accurate) than the SELECT COUNT(\*)?

Hints:

• This AW schema was created using the “innodb” database engine.

**With such a database engine, it can provide an estimated row count based on statistical sampling without scanning the entire table.**

**SELECT COUNT(\*) locks and scans the whole table, which takes more time to compute the exact number of rows.**

**And information\_schema is doing an estimation. It is less-accuracy but more time-efficiency.**

2. Use the information\_schema to list out each table in the adventureworks data warehouse and

its primary key.

Hints:

• There is a table within information\_schema called COLUMNS.

• Look for a column called column\_key

Graphical user interface, application, table

Description automatically generated with medium confidence

3. What standard table naming convention did the AdventureWorksDW database designers use

to differentiate dimension tables from fact tables in this star schema data warehouse?

**All the dimension tables have been prefixed with “Dim”**

**All the fact tables have been prefixed with “Fact”**

**In this star schema data warehouse, the prefix differentiates dimension tables from fact tables.**

4. What is the purpose of the recursive relationship on DimEmployee?

**The purpose of the recursive relationship is to represent the hierarchical relationship between employees. Employee can report to their boss, who is another employee shown in the table. In other words, the recursive relationship also reflects the parent-child relationship between employees.**

5. What are the three types of models of bikes sold by AdventureWorks? Provide your SQL

query, and your answer set along with your answer to the question. (HINT: There are many

models, but all those models fall into just three major types of bikes.)

Mountain bikes, Road Bikes, Touring Bikes

**SELECT DISTINCT EnglishProductSubcategoryName**

**FROM DimProductSubcategory**

**INNER JOIN DimProductCategory**

**ON DimProductSubcategory.ProductCategoryKey = DimProductCategory.ProductCategoryKey**

**WHERE DimProductCategory.EnglishProductCategoryName = 'Bikes';**

**Graphical user interface, text, application

Description automatically generated**

6. Of these three, which type of bike model had the highest sales (in dollar volume) in 2004?

Provide your SQL query, and your answer set along with your answer to the question.

Mountain Bikes has the highest sales.

**SELECT DISTINCT EnglishProductSubcategoryName, SUM(FactInternetSales.SalesAmount) AS 'TotalSales'**

**FROM DimProductSubcategory**

**INNER JOIN DimProduct**

**ON DimProductSubcategory. ProductSubcategoryKey = DimProduct.ProductSubcategoryKey**

**INNER JOIN FactInternetSales**

**ON DimProduct.ProductKey = FactInternetSales.ProductKey**

**INNER JOIN DimTime**

**ON FactInternetSales.OrderDateKey = DimTime.TimeKey**

**WHERE DimProductSubcategory.ProductCategoryKey = '1'**

**AND**

**DimTime.CalendarYear = '2004'**

**GROUP BY EnglishProductSubcategoryName;**

**Graphical user interface, application

Description automatically generated with medium confidence**

7. List six of the other non-bike products sold by AdventureWorks. (Pick any six.) Provide

your SQL query, and your answer set along with your answer to the question.

**SELECT EnglishProductSubcategoryName**

**FROM DimProductSubcategory**

**WHERE ProductCategoryKey != 1**

**LIMIT 6;**

**Table

Description automatically generated**

8. Compare and rank the total counts of the bikes sold by AdventureWorks for each of the years

2001 – 2004 by color. What was the most popular color of bikes sold in each of these 4

years? Provide your SQL query, and your answer set along with your answer to the question.

You can assume that one row in the fact table equals one sale.

**Red bike is the best sold of 2001**

**Red bike is the best sold of 2002**

**Black bike is the best sold of 2001**

**Black bike is the best sold of 2001**

**SELECT DISTINCT DimProduct.Color, COUNT(FactInternetSales.OrderQuantity) AS 'TotalBikeSales', DimTime.CalendarYear**

**FROM DimProductSubcategory**

**INNER JOIN DimProduct**

**ON DimProductSubcategory. ProductSubcategoryKey = DimProduct.ProductSubcategoryKey**

**INNER JOIN FactInternetSales**

**ON DimProduct.ProductKey = FactInternetSales.ProductKey**

**INNER JOIN DimTime**

**ON FactInternetSales.OrderDateKey = DimTime.TimeKey**

**WHERE DimProductSubcategory.ProductCategoryKey = '1'**

**AND DimTime.CalendarYear BETWEEN 2001 AND 2004**

**GROUP BY DimProduct.Color,DimTime.CalendarYear**

**ORDER BY DimTime.CalendarYear,**

**COUNT(FactInternetSales.OrderQuantity) DESC;**

**Table

Description automatically generated**

HINT: Since the fact table contains sales for all kinds of products, you should include only

fact rows where the sale is for a bike. One easy way to do this is a WHERE clause selecting

only rows where EnglishProductSubcategoryName contains the string “bikes”. There are

other ways to determine this as well.

9. List and compare the total sales quantities (number of bikes, NOT dollars) of bikes sold (all

model types) by customer gender by year and month. In which year and month were bike

sales to females the highest? Provide your SQL query, and your answer set along with your

answer to the question.

June, 2004 has the highest sales.

**SELECT DISTINCT COUNT(\*) AS 'TotalBikeSales', DimTime.CalendarYear, DimTime.EnglishMonthName, DimCustomer.Gender**

**FROM DimProductSubcategory**

**INNER JOIN DimProduct**

**ON DimProductSubcategory. ProductSubcategoryKey = DimProduct.ProductSubcategoryKey**

**INNER JOIN FactInternetSales**

**ON DimProduct.ProductKey = FactInternetSales.ProductKey**

**INNER JOIN DimCustomer**

**ON DimCustomer.CustomerKey = FactInternetSales.CustomerKey**

**INNER JOIN DimTime**

**ON FactInternetSales.OrderDateKey = DimTime.TimeKey**

**WHERE DimCustomer.Gender = 'F'**

**AND DimProductSubcategory.ProductCategoryKey = '1'**

**GROUP BY DimCustomer.Gender, DimTime.CalendarYear, DimTime.EnglishMonthName**

**ORDER BY COUNT(\*) DESC;**

**Table

Description automatically generated**

10. For the year 2004, which State/Province yielded the highest margin for AdventureWorks?

(HINT: use the customer’s State/Province.) Provide your SQL query, and your answer set

along with your answer to the question. Margin is the difference between what

Adventureworks PAID for a bike and what they SOLD it for.

New South Wales yield the highest margin for AdventureWorks

SELECT DISTINCT SUM((FactInternetSales.UnitPrice - FactInternetSales.ProductStandardCost) ) AS 'highest margin', DimTime.CalendarYear, DimGeography.StateProvinceName AS Province

FROM DimProductSubcategory

INNER JOIN DimProduct

ON DimProductSubcategory. ProductSubcategoryKey = DimProduct.ProductSubcategoryKey

INNER JOIN FactInternetSales

ON DimProduct.ProductKey = FactInternetSales.ProductKey

INNER JOIN DimCustomer

ON DimCustomer.CustomerKey = FactInternetSales.CustomerKey

INNER JOIN DimTime

ON FactInternetSales.OrderDateKey = DimTime.TimeKey

INNER JOIN DimGeography

On DimGeography.GeographyKey = DimCustomer.GeographyKey

WHERE DimTime.CalendarYear = '2004'

AND DimProductSubcategory.ProductCategoryKey = '1'

GROUP BY DimGeography.StateProvinceName

ORDER BY 'highest margin' DESC

LIMIT 10;

Table

Description automatically generated