Free Questions for QSDA2024

Shared by Gonzales on 04-10-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

A company's analytics team is migrating from QlikView to Qlik Sense. During the transition there is an opportunity to improve overall reporting.

Which set of criteria must the data architect consider while planning for the migration?

Options:

A- Application size, application theme, storytelling, data model, IT use case

B- User sessions, source data architecture, compatibility, data model, business use case

C- QlikView archival, source data architecture, load script, data model, business use case

D- Application metadata, application theme, user sessions, load script, IT use case

Answer:

C

Explanation:

During the transition from QlikView to Qlik Sense, the analytics team has the opportunity to improve the overall reporting. To ensure a smooth migration while optimizing the new environment, the data architect needs to consider several key factors.

Option C is the best choice because it encompasses the essential aspects of a migration project:

QlikView Archival:

Archiving QlikView applications is crucial to ensure that historical data and applications are preserved and can be referenced if needed in the future. This step is important to maintain continuity and provide a fallback option if required during the transition.

Source Data Architecture:

Understanding the existing source data architecture is critical to ensure that the new Qlik Sense applications can seamlessly connect to the data sources. This also helps in identifying opportunities to optimize or re-architect the data pipelines for better performance in Qlik Sense.

Load Script:

The load script from QlikView might need to be revised or optimized for Qlik Sense. It's important to ensure that the script is compatible and takes advantage of Qlik Sense's capabilities, such as improved data handling, better inline transformations, and enhanced scripting functions.

Data Model:

Reviewing and possibly redesigning the data model is essential during the migration. Qlik Sense's associative engine allows for more flexibility, and this is an opportunity to improve the data model for better performance, scalability, and user experience.

Business Use Case:

Understanding the business use case is vital to ensure that the new Qlik Sense applications meet the business requirements effectively. This includes making sure that the new reports and dashboards are aligned with the business goals and provide the necessary insights.

Qlik Migration Guide: When migrating from QlikView to Qlik Sense, it's important to consider not just the technical aspects but also the business implications and opportunities for improvement.

Qlik Documentation on Data Modeling and Load Script Optimization: These resources provide best practices on how to optimize load scripts and data models during migration to ensure smooth operation and better performance in Qlik Sense.

Question 2

Question Type: MultipleChoice

A company generates I GB of ticketing data daily. The data is stored in multiple tables. Business users need to see trends of tickets processed for the past 2 years. Users very rarely access the transaction-level data for a specific date. Only the past 2 years of data must be loaded, which is 720 GB of data.

Which method should a data architect use to meet these requirements?

Options:

A- Load only 2 years of data in an aggregated app and create a separate transaction app for occasional use

B- Load only 2 years of data and use best practices in scripting and visualization to calculate and display aggregated data

C- Load only aggregated data for 2 years and use On-Demand App Generation (ODAG) for transaction data

D- Load only aggregated data for 2 years and apply filters on a sheet for transaction data

Answer:

C

Explanation:

In this scenario, the company generates 1 GB of ticketing data daily, accumulating up to 720 GB over two years. Business users mainly require trend analysis for the past two years and rarely need to access the transaction-level data. The objective is to load only the necessary data while ensuring the system remains performant.

Option C is the optimal choice for the following reasons:

Efficiency in Data Handling:

By loading only aggregated data for the two years, the app remains lean, ensuring faster load times and better performance when users interact with the dashboard. Aggregated data is sufficient for analyzing trends, which is the primary use case mentioned.

On-Demand App Generation (ODAG):

ODAG is a feature in Qlik Sense designed for scenarios like this one. It allows users to generate a smaller, transaction-level dataset on demand. Since users rarely need to drill down into transaction-level data, ODAG is a perfect fit. It lets users load detailed data for specific dates only when needed, thus saving resources and keeping the main application lightweight.

Performance Optimization:

Loading only aggregated data ensures that the application is optimized for performance. Users can analyze trends without the overhead of transaction-level details, and when they need more detailed data, ODAG allows for targeted loading of that data.

Qlik Sense Best Practices: Using ODAG is recommended when dealing with large datasets where full transaction data isn't frequently needed but should still be accessible.

Qlik Documentation on ODAG: ODAG helps in maintaining a balance between performance and data availability by providing a method to load only the necessary details on demand.
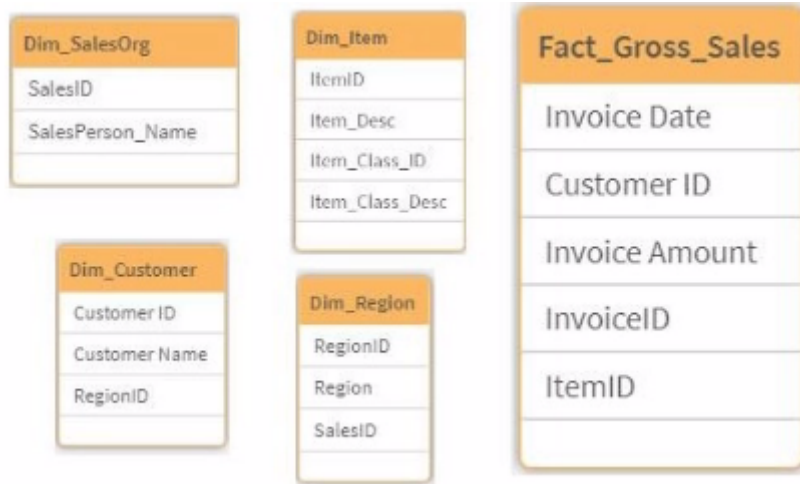
Question 3

Question Type: **MultipleChoice**

Exhibit.

Dim_SalesOrg
- SalesID
- SalesPerson_Name

Dim_Item
- ItemID
- Item_Desc
- Item_Class_ID
- Item_Class_Desc

Dim_Customer
- Customer ID
- Customer Name
- RegionID

Dim_Region
- RegionID
- Region
- SalesID

Fact_Gross_Sales
- Invoice Date
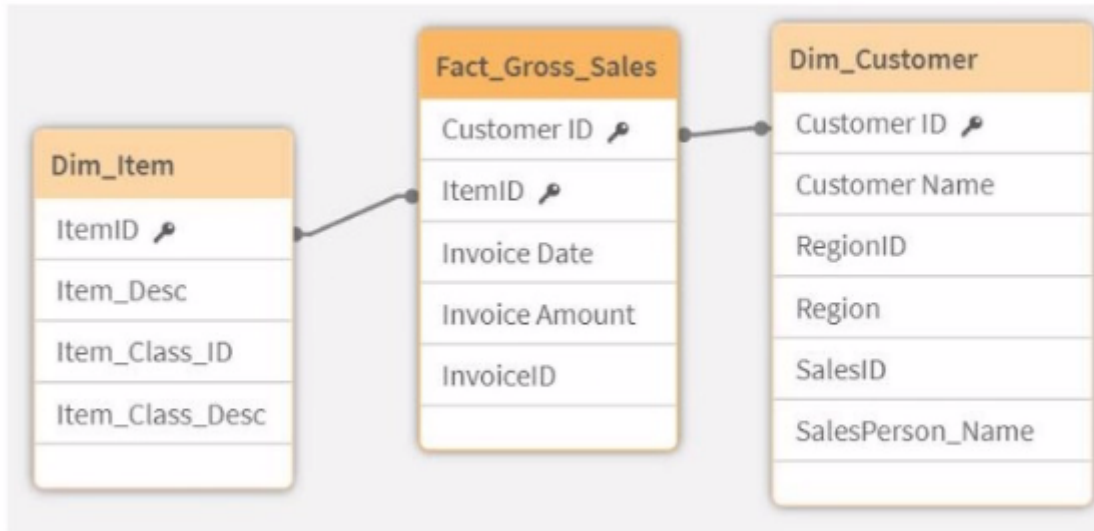- Customer ID
- Invoice Amount
- InvoiceID
- ItemID

Refer to the exhibit.

A data architect is provided with five tables. One table has Sales Information. The other four tables provide attributes that the end user will group and filter by.
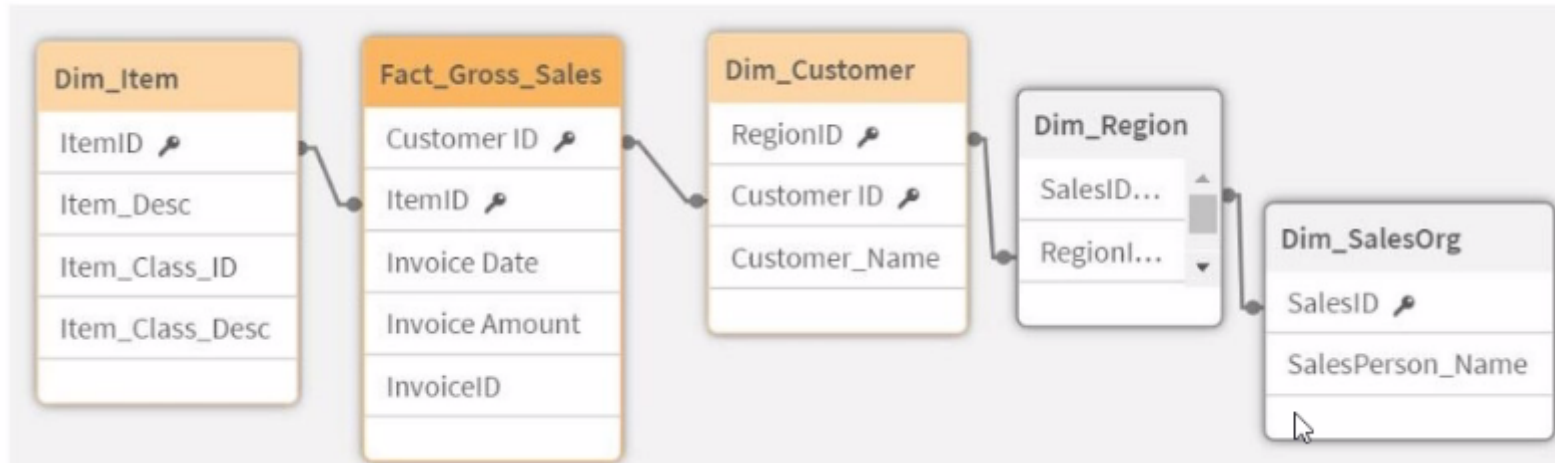
There is only one Sales Person in each Region and only one Region per Customer.

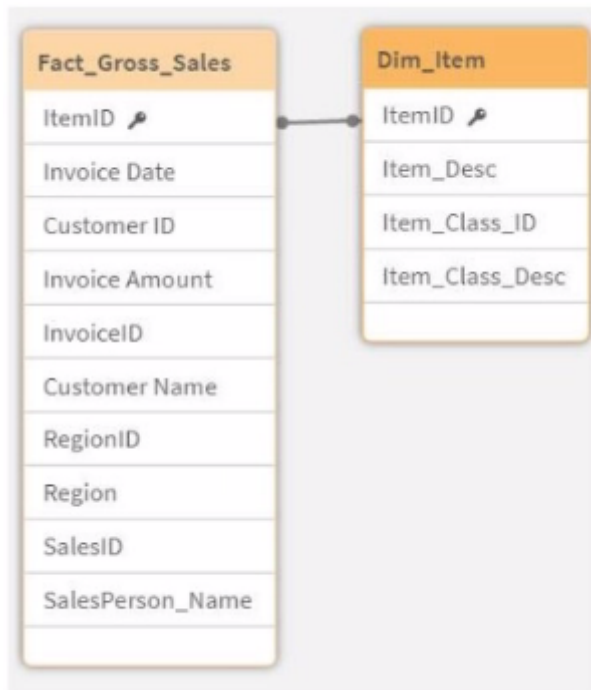Which data model is the most optimal for use in this situation?
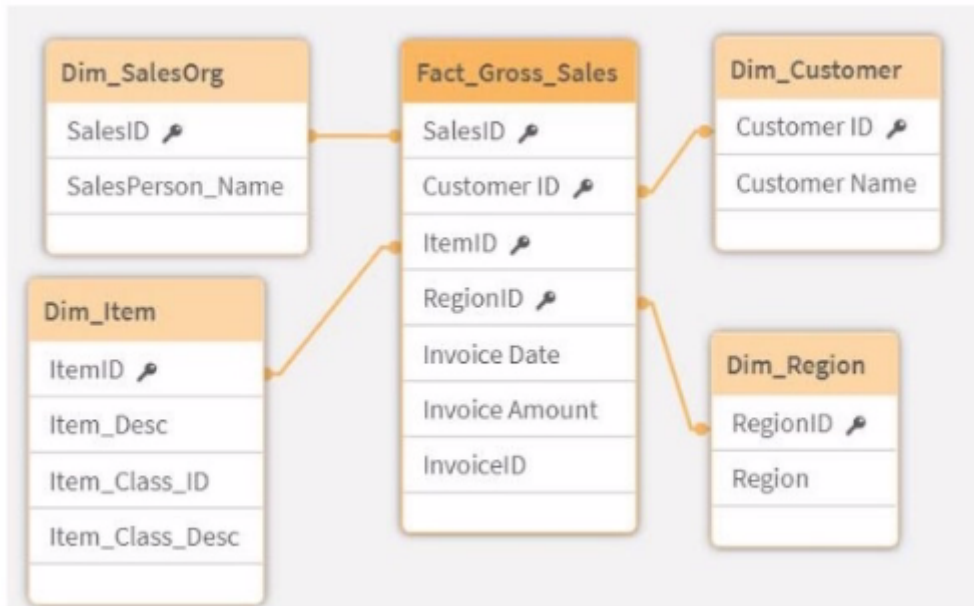
A)

B)

C)

Fact_Gross_Sales
- ItemID 🔑
- Invoice Date
- Customer ID
- Invoice Amount
- InvoiceID
- Customer Name
- RegionID
- Region
- SalesID
- SalesPerson_Name

Dim_Item
- ItemID 🔑
- Item_Desc
- Item_Class_ID
- Item_Class_Desc

D)

Options:

A- Option A

B- Option B

C- Option C

D- Option D

Answer:

D

Explanation:

In the given scenario, where the data architect is provided with five tables, the goal is to design the most optimal data model for use in Qlik Sense. The key considerations here are to ensure a proper star schema, minimize redundancy, and ensure clear and efficient relationships among the tables.

Option D is the most optimal model for the following reasons:

Star Schema Design:

In Option D, the Fact_Gross_Sales table is clearly defined as the central fact table, while the other tables (Dim_SalesOrg, Dim_Item, Dim_Region, Dim_Customer) serve as dimension tables. This layout adheres to the star schema model, which is generally recommended in Qlik Sense for performance and simplicity.

Minimization of Redundancies:

In this model, each dimension table is only connected directly to the fact table, and there are no unnecessary joins between dimension tables. This minimizes the chances of redundant data and ensures that each dimension is only represented once, linked through a unique key to the fact table.

Clear and Efficient Relationships:

Option D ensures that there is no ambiguity in the relationships between tables. Each key field (like Customer ID, SalesID, RegionID, ItemID) is clearly linked between the dimension and fact tables, making it easy for Qlik Sense to optimize queries and for users to perform accurate aggregations and analysis.

Hierarchical Relationships and Data Integrity:

This model effectively represents the hierarchical relationships inherent in the data. For example, each customer belongs to a region, each salesperson is associated with a sales organization, and each sales transaction involves an item. By structuring the data in this way, Option D maintains the integrity of these relationships.

Flexibility for Analysis:

The model allows users to group and filter data efficiently by different attributes (such as salesperson, region, customer, and item). Because the dimensions are not interlinked directly with each other but only through the fact table, this setup allows for more flexibility in creating visualizations and filtering data in Qlik Sense.

Qlik Sense Best Practices: Adhering to star schema designs in Qlik Sense helps in simplifying the data model, which is crucial for performance optimization and ease of use.

Data Modeling Guidelines: The star schema is recommended over snowflake schema for its simplicity and performance benefits in Qlik Sense, particularly in scenarios where clear relationships are essential for the integrity and accuracy of the analysis.

Question 4

Question Type: MultipleChoice

Exhibit.

Dynamic dimension data				Static dimension data	
SPID	**ChangeDate**	**Department**		**SPID**	**Name**
1		Dept B		1	Bob
2		Dept C		2	Cynthia
2	12/31/2011	Dept D			

A large electronics company re-assigns sales people once per year from one Department to another.

SPID is the Salesperson ID; the SPID for each individual sales person Name remains constant. The Department for a SPID may change; each change is stored in the Dynamic Dimension data.

Four tables need to be linked correctly: a transaction table, a dynamic salesperson dimension, a static salesperson dimension, and a department dimension.

Which script prefix should the data architect use?

Options:

A- Merge

B- IntervalMatch

C- Partial Reload

D- Semantic

Answer:

B

Explanation:

In the scenario described, the Dynamic Dimension data tracks changes in department assignments for salespeople over time. To correctly link the transaction data with the salesperson data and ensure that sales are associated with the correct department based on the date, an IntervalMatch function should be used.

IntervalMatch is designed to match discrete data (like transaction dates) with a range of dates. In this case, each salesperson's department assignment is valid over a period of time, and the IntervalMatch function can be used to link the transaction data with the correct department for each salesperson based on the transaction date.

Option A (Merge): This option is incorrect as it refers to combining data sets, which doesn't address the need to handle the dynamic, date-based department assignments.

Option B (IntervalMatch): This is the correct choice because it allows you to match each transaction with the correct department assignment based on the ChangeDate in the Dynamic Dimension data.

Option C (Partial Reload): This refers to reloading only part of the data, which is not relevant to linking tables based on date ranges.

Option D (Semantic): This option is not applicable as it refers to a broader approach to data modeling and interpretation rather than specifically linking data based on time intervals.

Thus, IntervalMatch is the correct method for linking the transaction data with the dynamic salesperson dimension, ensuring that each transaction is associated with the correct department based on the historical assignment data.

Question 5

Refer to the exhibit.

A system creates log files and csv files daily and places these files in a folder. The log files are named automatically by the source system and change regularly. All csv files must be loaded into Qlik Sense for analysis.

Which method should be used to meet the requirements?

A)

```
SET vFileList = 'Day1','Day2','Day3';
FOR EACH vFile in $(vFileList)
LOAD * FROM [lib://Data (User1)/multi/$(vFile).csv];
NEXT
```

B)

```
LOAD *
FROM [lib://Data (User1)/multi/*]
(csv);
```

C)

```
LOAD *
FROM [lib://Data (User1)/multi/*]
(txt);
```

D)

```
SET vLib = chr(39) & 'lib://Data (User1)/multi/*.csv' & chr(39);
FOR EACH File in filelist($(vLib))
LOAD * FROM [$(File)];
NEXT
```

Options:

A- Option A

B- Option B

C- Option C

D- Option D

Answer:

B

Explanation:

In the scenario described, the goal is to load all CSV files from a directory into Qlik Sense, while ignoring the log files that are also present in the same directory. The correct approach should allow for dynamic file loading without needing to manually specify each file name, especially since the log files change regularly.

Here's why Option B is the correct choice:

Option A: This method involves manually specifying a list of files (Day1, Day2, Day3) and then iterating through them to load each one. While this method would work, it requires knowing the exact file names in advance, which is not practical given that new files are added regularly. Also, it doesn't handle dynamic file name changes or new files added to the folder automatically.

Option B: This approach uses a wildcard (*) in the file path, which tells Qlik Sense to load all files matching the pattern (in this case, all CSV files in the directory). Since the csv file extension is explicitly specified, only the CSV files will be loaded, and the log files will be ignored. This method is efficient and handles the dynamic nature of the file names without needing manual updates to the script.

Option C: This option is similar to Option B but targets text files (txt) instead of CSV files. Since the requirement is to load CSV files, this option would not meet the needs.

Option D: This option uses a more complex approach with filelist() and a loop, which could work, but it's more complex than necessary. Option B achieves the same result more simply and directly.

Therefore, Option B is the most efficient and straightforward solution, dynamically loading all CSV files from the specified directory while ignoring the log files, as required.

Question 6

Question Type: MultipleChoice

Refer to the exhibit.

Sales	Customers	Employees
SaleID	CustID	EmployeeID
CustomerID	CustName	EmployeeName
Amount	Address	MgrID
SaleDate	City	
SalesPersonID	State	
RegionalAcctMgrID	Zip	

Refer to the exhibit.

A data architect needs to create a data model for a new app. Users must be able to see:

* Total sales for each customer

* Total sales for a given state

* Customers that have not had any sales

* Names of salesperson and regional account managers

* Total number of sales by date

Which steps should the data architect perform to meet these requirements?

Which steps should the data architect perform to meet these requirements?

Options:

A- 1. Use a Mapping Load for the Employees table

2. Load the Sales table and use ApplyMap to get the names for SalesPersonID and RegionalAcctMgrID

3. Use a Left Join Load to add the customer details for the Sales table

B- 1. Load the Customers table and alias the CustID field as CustomerID

2. Use a Mapping Load for the Employees table

3. Load the Sales table and use ApplyMap to get the names for SalesPersonID and RegionalAcctMgrID

C- 1. Load the Sales table

2. Load the Customers table

3. Load the Employees table twice; name it and alias the EmployeeID field appropriately each time

D- 1. Load the Customers table and alias the CustID field as CustomerID

2. Load the Employees table

3. Load the Sales table and alias the SalesPersonID and RegionalAcctMgrID fields as EmployeeID

Answer:

C

Explanation:

In the provided scenario, the data architect needs to create a data model that supports various analyses, including total sales for each customer, total sales by state, identifying customers with no sales, and displaying the names of salespersons and regional account managers.

Here's why Option C is the correct choice:

Loading the Sales Table: The Sales table contains key information related to sales transactions, including SaleID, CustomerID, Amount, SaleDate, SalesPersonID, and RegionalAcctMgrID. This table must be loaded first as it will be central to the analysis.

Loading the Customers Table: The Customers table includes customer details such as CustID, CustName, Address, City, State, and Zip. Loading this table and linking it to the Sales table via the CustomerID field allows you to perform analyses such as total sales per customer and total sales by state. Importantly, loading the customers separately will also allow the identification of customers without any sales.

Loading the Employees Table Twice: The Employees table must be loaded twice because it is used to look up two different roles in the sales process: the SalesPersonID and the RegionalAcctMgrID. When loading the table twice:

The first instance of the Employees table will be used to map the SalesPersonID to EmployeeName.

The second instance will be used to map the RegionalAcctMgrID to EmployeeName.

Aliasing the EmployeeID field appropriately in each instance is crucial to prevent creating synthetic keys and to ensure the correct association with the roles in the sales process.

This approach ensures that the data model will correctly support all the required analyses, including identifying customers without sales, which is crucial for meeting the business requirements.

Option A and Option B propose using a mapping load and ApplyMap, which can complicate the model and does not directly address all the business requirements.

Option D involves aliasing fields in a way that could create unnecessary complexity and might not accurately reflect the relationships in the data.

Thus, Option C is the correct answer as it best meets the requirements while maintaining a clear and functional data model.

Question 7

A data architect in the Enterprise Architecture team wants to develop a new application summarizing Qlik Sense usage by all company employees. They also want to gather usage metrics for other systems.

Who should the data architect contact to be granted access to the data?

Options:

A- IT Security Director, Human Resources Director, Qlik Sense Administrator

B- IT Security Manager, Qlik Sense Account Manager, Enterprise Architecture Director

C- IT Security Analyst, Qlik Sense Developers, Solutions Architect

D- IT Security Vice President, Human Resources Analyst, Qlik Sense Developers

Answer:

A

Explanation:

When developing an application that summarizes Qlik Sense usage by company employees and also gathers usage metrics for other systems, the data architect needs to ensure they have the correct access to sensitive data. The following roles are crucial:

IT Security Director: Responsible for the security of IT systems and data. They would ensure that the data architect has the appropriate permissions to access usage metrics and other system data securely.

Human Resources Director: They manage employee-related data, including employment records that might be necessary for matching employee IDs with usage metrics. This access is crucial for correlating usage data with specific employees.

Qlik Sense Administrator: This individual has administrative rights over the Qlik Sense environment and can grant access to usage data within Qlik Sense, ensuring that the architect has the necessary data to analyze.

Given the need to securely and correctly handle sensitive data, including employee usage metrics across multiple systems, Option A includes all the appropriate contacts for access and permissions.