

DUMPSBOSS.

SnowPro Associate: Platform Certification Exam

Snowflake SOL-C01

Version Demo

Total Demo Questions: 18

Total Premium Questions: 183

Buy Premium PDF

<https://dumpsboss.co>

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, Snowflake AI Data Cloud Features and Architecture	79
Topic 2, Data Movement	34
Topic 3, Data Analysis	29
Topic 4, Data Governance and Administration	41
Total	183

QUESTION NO: 1

Which of the following is a potential use case for the CLASSIFY_TEXT function?

- A. Generating SQL code from natural language description
- B. Categorizing customer support tickets based on topic
- C. Translating a user manual into multiple languages
- D. Extracting data from scanned invoices

ANSWER: B

Explanation:

CLASSIFY_TEXT is designed to assign unstructured text to one of a defined set of categories. Therefore, **Categorizing customer support tickets based on topic** is a direct and practical use case. For example, an organization can supply category labels such as “billing,” “account access,” “technical issue,” and “feature request,” then classify each incoming ticket according to its content. The result can be used to route tickets to the appropriate queue, prioritize particular issue types, measure support trends, and reduce manual triage effort.

Snowflake Cortex classification uses examples associated with categories to create a classification model, and CLASSIFY_TEXT then returns the category prediction for each text input. This makes it suitable for repeated business classification tasks involving free-form text, including support tickets, feedback, reviews, survey responses, and message streams. The function can be invoked in SQL, allowing the classification output to be incorporated into Snowflake data pipelines and analytics workflows without exporting the data for separate processing. Snowflake documents the function and its category-based model workflow at [Snowflake Cortex classification](#) and provides SQL syntax and result details in the [CLASSIFY_TEXT function reference](#).

QUESTION NO: 2

What is a benefit of using an external stage to load data into Snowflake?

- A. External stages reduce data storage costs because data is stored outside Snowflake.
- B. External stages provide automatic data purging after successful loads.
- C. External stages are more secure than internal stages for sensitive data.
- D. External stages reduce the number of objects in a database.

ANSWER: A

Explanation:

External stages reduce data storage costs because data is stored outside Snowflake. An external stage stores the location and access configuration for files held in a supported cloud-storage service, such as Amazon S3, Google Cloud Storage, or Microsoft Azure Blob Storage. Snowflake can read files from that location during a COPY INTO operation without requiring those source files to be uploaded first into a Snowflake internal stage. Consequently, the staged source files do not consume Snowflake-managed storage while they remain in the external location; their storage charges are instead governed by the applicable cloud-storage provider and account configuration. This can be especially useful when an organization already maintains a data lake or landing zone in cloud object storage, or when it needs to retain large source files outside Snowflake for operational, archival, or sharing purposes. Snowflake documentation describes external stages as references to external cloud storage locations and distinguishes them from internal stages, where files are stored in Snowflake. See Snowflake's [overview of creating stages](#) and its [stage considerations documentation](#) for details on external storage integrations, supported locations, and loading behavior.

QUESTION NO: 3

Which types of stages are supported in Snowflake for data loading and unloading? (Select TWO)

- A. Managed Stages
- B. Permanent Stages
- C. Temporary Stages
- D. External Stages
- E. Internal Stages

ANSWER: D E

Explanation:

Snowflake supports **External Stages** and **Internal Stages** for loading data into Snowflake tables and unloading query or table results. Internal stages store files in Snowflake-managed storage. They include user stages, table stages, and named internal stages, and are useful when files should be uploaded directly to Snowflake before executing loading operations such as `COPY INTO`. Snowflake manages the underlying cloud-storage location for these stages.

External stages point to files in externally managed cloud storage. Snowflake supports external locations in Amazon S3, Google Cloud Storage, and Microsoft Azure Blob Storage. A named external stage stores the cloud location along with access configuration, such as a storage integration or credentials, allowing the location to be reused for loading and unloading workflows. Both internal and external stages can be referenced by Snowflake data-loading and data-unloading commands, with the applicable permissions and file-format settings.

Snowflake documentation describes stages according to whether files are stored internally by Snowflake or externally in a supported cloud-storage provider. See [Snowflake documentation on internal stages](#) and [Snowflake documentation on external stages](#).

QUESTION NO: 4

What is the highest level object in the Snowflake object hierarchy?

- A. Database
- B. Virtual Warehouse
- C. Schema
- D. Account

ANSWER: D

Explanation:

Account is the highest-level container in Snowflake's object hierarchy. A Snowflake account provides the overall boundary in which an organization's Snowflake resources, security administration, and data objects are managed. Within an account, databases contain schemas, and schemas contain objects such as tables, views, stages, and functions. Virtual warehouses are also account-level objects that provide compute resources for executing queries and other operations, but they are created and managed within the account rather than serving as the container for it.

The account is also the scope for many important administrative and governance capabilities, including users, roles, warehouses, resource monitors, integrations, and account-level parameters. Snowflake documentation describes the account as the top-level organizational unit and distinguishes account objects from schema objects. This hierarchy is important because object names, ownership, privileges, and administration are evaluated within the context of a Snowflake account. For more information, see Snowflake's [Snowflake overview](#) and its [object hierarchy documentation](#).

QUESTION NO: 5

Which of the following are date and time data types in Snowflake? (Choose any 3 options)

- A. TIMEDATE
- B. DATE
- C. Python
- D. TIMESTAMP
- E. DATETIME
- F. JavaScript

ANSWER: B D E

Explanation:

DATE, **TIMESTAMP**, and **DATETIME** are valid Snowflake date and time data types. **DATE** stores a calendar date consisting of a year, month, and day, without a time-of-day component. It is appropriate for values such as order dates, birthdays, and reporting periods where time is not needed.

TIMESTAMP stores both a date and a time. In Snowflake, **TIMESTAMP** is a user-facing alias that resolves to one of Snowflake's timestamp variants according to the `TIMESTAMP_TYPE_MAPPING` session parameter. The underlying variants support timestamps with no time zone, local time zone handling, or an explicitly stored offset and time zone.

DATETIME is also supported by Snowflake as an alias for `TIMESTAMP_NTZ`. Consequently, it stores a date and time without time-zone information. This is useful when a timestamp should be interpreted as a literal wall-clock value rather than converted based on a session time zone. Snowflake documents **DATE**, **TIME**, and its **TIMESTAMP** forms in its date and time data-type reference, including the **DATETIME** alias. See the [Snowflake date and time data types documentation](#) and the [TIMESTAMP_TYPE_MAPPING parameter reference](#).

QUESTION NO: 6

Which of the following are benefits of using a multi-cluster warehouse in Snowflake? (Choose any 2 options)

- A. Reduced storage costs
- B. Improved concurrency for handling many simultaneous queries
- C. Automatic scaling to meet varying query loads
- D. Faster data loading

ANSWER: B C

Explanation:

Multi-cluster warehouses are designed to support high-concurrency workloads. "Improved concurrency for handling many simultaneous queries" is a key benefit because Snowflake can route work across multiple compute clusters that operate under one logical warehouse. This helps reduce or prevent queuing when many users, dashboards, or applications submit queries at the same time. It is particularly useful for mixed BI and reporting workloads where the demand for compute can rise sharply and unpredictably.

"Automatic scaling to meet varying query loads" is also a benefit. When configured in Auto-scale mode, Snowflake starts additional clusters as workload demand increases, up to the warehouse's configured maximum cluster count. As demand falls, it can shut down unneeded clusters, subject to the scaling policy. This provides elastic concurrency capacity while allowing organizations to control the minimum and maximum number of active clusters. Multi-cluster warehouses therefore address workload contention through horizontal scaling, rather than changing the size of a single cluster. See Snowflake's [Multi-cluster warehouses](#) documentation and its guidance on [virtual warehouses](#) for configuration and operational details.

QUESTION NO: 7

Which of the following are valid Snowflake Virtual Warehouse sizes? (Choose any 3 options)

- A. Small
- B. Tiny
- C. Medium
- D. X-Small

ANSWER: A C D

Explanation:

Snowflake virtual warehouses use a defined set of compute-size tiers. **Small**, **Medium**, and **X-Small** are all valid sizes and are the three applicable choices in this question. X-Small is Snowflake's smallest standard warehouse tier and is suitable for lighter workloads, such as development, ad hoc queries, or relatively low-volume processing. Small is the next available tier, and Medium provides another increase in compute capacity for workloads requiring greater concurrency or faster execution.

Warehouse sizing is central to Snowflake performance and cost management because the selected tier determines the compute resources available to execute queries and other workloads. Administrators can create warehouses using these predefined sizes and can resize a warehouse as workload requirements change. Snowflake documents the available size values in its warehouse-creation syntax, including X-Small, Small, Medium, Large, X-Large, and larger tiers. The documentation also describes warehouse considerations, including how size relates to compute resources and scaling decisions. See the [CREATE WAREHOUSE reference](#) and Snowflake's [virtual warehouse overview](#).

QUESTION NO: 8

Which of the following components are included in a fully qualified name in Snowflake? (Choose any 3 options)

- A. Database name
- B. Role name
- C. Schema name
- D. Table or view name

ANSWER: A C D

Explanation:

In Snowflake, a fully qualified object name uses the three-part namespace format `database_name.schema_name.object_name`. Therefore, **Database name**, **Schema name**, and **Table or view name** are the requested components. The database identifies the top-level container in which the object resides, while the schema identifies the namespace within that database. The final element is the object identifier itself; for this question, that can be a table or a view. For example, `SALES_DB.PUBLIC.ORDERS` identifies the `ORDERS` table in the `PUBLIC` schema of the `SALES_DB` database.

Using all three parts makes a reference explicit and independent of the session's current database and schema settings. This is particularly useful in shared environments, deployment scripts, views, and cross-database queries, where identically named objects may exist in more than one schema or database. Snowflake documentation describes the fully qualified form for database objects as `database.schema.object`; it also supports partially qualified and unqualified references when the applicable session context is set. See Snowflake's [object name resolution documentation](#) and its [identifier requirements documentation](#).

QUESTION NO: 9

Which edition of Snowflake offers the most comprehensive security and compliance features, including HIPAA and PCI DSS compliance?

- A. Standard Edition
- B. Business Critical Edition
- C. Enterprise Edition
- D. Virtual Private Snowflake (VPS) Edition

ANSWER: B

Explanation:

Business Critical Edition is the Snowflake service edition designed for organizations that manage highly sensitive data or operate in regulated industries. It includes the capabilities of lower editions and adds enhanced security features and compliance support appropriate for stringent regulatory requirements. In particular, Snowflake identifies Business Critical as the edition for customers that require support for HIPAA and PCI DSS compliance. These frameworks place strong requirements on the protection, handling, access control, and auditing of health-related and payment-card data, so this edition is the appropriate selection when those needs are stated in the question.

Business Critical also provides additional platform protections, including features such as Tri-Secret Secure, which enables a customer-managed key to be combined with Snowflake-managed key material for encryption. Snowflake's editions documentation describes Business Critical as intended for organizations with sensitive data and rigorous security requirements, while its security and compliance materials list the compliance programs supported by the platform. Virtual Private Snowflake is an additional isolation offering available for qualifying Business Critical deployments; it does not replace Business Critical as the edition associated with HIPAA and PCI DSS support. See Snowflake's [service editions overview](#) and [security and compliance overview](#).

QUESTION NO: 10

Which of the following statements are true about Zero-Copy Cloning in Snowflake? (Select TWO)

- A. It requires significant additional storage.
- B. It is immediate and metadata-only operation.
- C. It duplicates the underlying data storage.
- D. It creates a writable copy of an object.

ANSWER: B D

Explanation:

Zero-Copy Cloning is immediate and metadata-only operation because Snowflake creates the clone by copying metadata that references existing micro-partitions rather than physically copying all underlying data. This makes cloning a fast way to create environments for development, testing, recovery, or experimentation, even when the source contains substantial data. Initially, the clone and its source share the data storage associated with unchanged micro-partitions. Snowflake tracks the clone independently after creation, so changes made to either object do not change the other object's logical contents. Accordingly, it creates a writable copy of an object: users can insert, update, delete, and otherwise modify supported cloned objects according to their granted privileges. Additional storage is charged only as data changes cause new micro-partitions to be written or retained independently; the initial cloning operation itself does not duplicate all physical data. Snowflake documents this behavior as metadata-based cloning with independent, writable clones and shared storage for unchanged data. See [Snowflake object cloning](#) and [Snowflake storage considerations for cloned tables](#).

QUESTION NO: 11

How do you specify a custom delimiter for a CSV file when using COPY INTO

?

Options:

A.
Using the FIELD_DELIMITER file format option

B.
Using the FIELD_SEPARATOR stage property.

C.
Using the DELIMITER_CHAR table property.

D.
Using the COLUMN_SEPARATOR query parameter

Hide Solution Discussion 0

Question # 6

What is the primary responsibility of Role Hierarchy in Snowflake's access control model?

Options:

A.
To allow roles to inherit privileges from parent roles

B.
To define the order of SQL command execution

C.
To categorize different types of users

D.
To specify the size of virtual warehouses

Hide Solution Discussion 0

Question # 7

Which statement is true about Snowflake Data Exchange? (Choose any 2 options)

Options:

A.
It is limited to internal data sharing only

B.
It requires complex ETL processes to transfer data

C.
It supports data sharing between different regions and cloud providers

D.
It allows organizations to securely share live, governed data

Suggested Solution Discussion 0

Question # 8

What is the purpose of the "auto-suspend" feature in Snowflake virtual warehouses?

Options:

A.

To automatically increase the size of the warehouse during peak load

B.

To automatically shut down the warehouse after a period of inactivity

C.

To automatically add more nodes to the warehouse

D.

To automatically reduce the number of clusters

Hide Solution Discussion 0

Question # 9

What Snowflake object provides a secure connection to external cloud storage?

Options:

A.

A directory table

B.

An external table

C.

An external stage

D.

A named file format

Hide Solution Discussion 0

Question # 10

What is the key difference between a "private listing" and a "public listing" in a Snowflake Data Exchange context?

Options:

A.

Public listings require payment, while private listings are free.

B.

Public listings are only accessible to Snowflake employees, while private listings are for external users.

C.

Private listings are shared with specific Snowflake accounts, while public listings are available to all Snowflake users.

D.

Private listings are for sharing code, while public listings are for sharing data.

Hide Solution Discussion 0

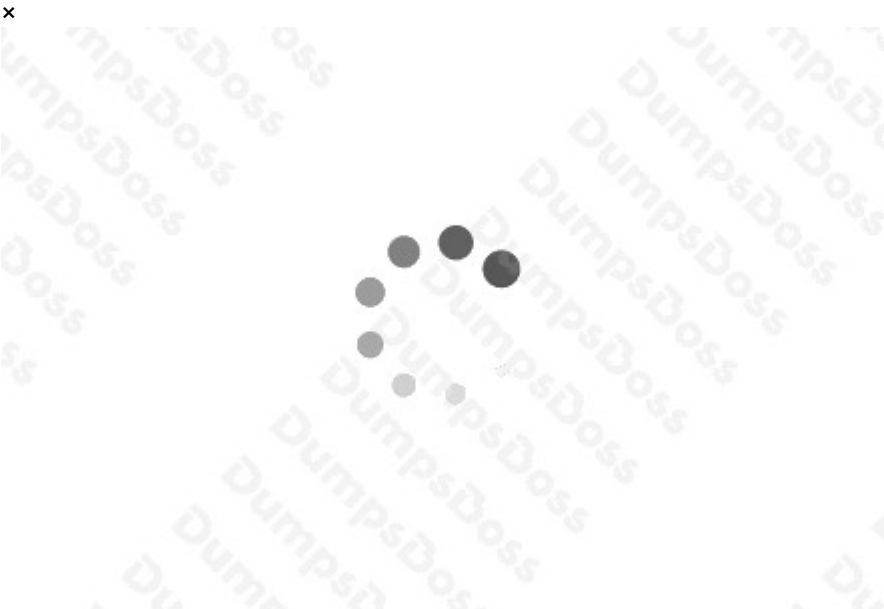
Prev

1

- 2
 - 3
 - 4
 - 5
 - 6
 - 7
- Next

Get SOL-C01 dumps and pass your exam in 24 hours!

Modal title



Registered Required

×
In order to participate in the comments you need to be logged-in.
You can sign-up or login (it's free).
OK

Free Exams Sample Questions

- Free SY0-701 Questions
- Free Rev-Con-201 Questions
- Free 2V0-13.25 Questions
- Free ZDTE Questions
- Free N10-009 Questions
- Free 200-301 Questions
- Free AZ-104 Questions
- Free 350-401 Questions
- Free 2V0-17.25 Questions
- Free CMMC-CCP Questions
- Free AIF-C01 Questions
- Free CMMC-CCA Questions
- Free AI-102 Questions
- Free 312-50v13 Questions
- Free ICWIM Questions
- Free CKA Questions
- Free SC-200 Questions
- Free SC-300 Questions
- Free OGEA-102 Questions
- NGFW-Engineer Questions

ISA-IEC-62443 Questions
Free Agentforce Specialist Questions
Free Generative-AI-Leader Questions
Free FCSS_EFW_AD-7.6 Questions
Free Secure-Software-Design Questions
Free Data-Cloud-Consultant Questions
Free Workday-Pro-Integrations Questions
Free UAE-Financial-Rules-and-Regulations Questions

We Accept



Secure Site



TESTED 09 Aug 2026

Customer Review

Hi this is Romona Kearns from Holland and I would like to tell you that I passed my exam with the use of exams4sure dumps. I got same questions in my exam that I prepared from your test engine software. I will recommend your site to all my friends for sure.



Romona Kearns Happy Customer

Money Back Guarantee

Our all material is important and it will be handy for you. If you have short time for exam so, we are sure with the use of it you will pass it easily with good marks. If you will not pass so, you could feel free to claim your refund. We will give 100% money back guarantee if our customers will not satisfy with our products.

AWS-Solution-Architect-Associate Dumps
Professional-Cloud-Architect Dumps
Service-Cloud-Consultant Dumps
Platform-App-Builder Dumps
AZ-500 Dumps
Associate-Cloud-Engineer Dumps
350-701 Dumps
ISO-IEC-27001-Lead-Implementer Dumps
Integration-Architect Dumps
CLF-C02 Dumps
CDCP Dumps
CTA Dumps
NCP-MCI-6.10 Dumps
NCA-AIIO Dumps

Home
About Us
FAQs

Copyright © 2014-2026 Exams4sure. All Rights Reserved

- A. Using the FIELD_DELIMITER file format option
- B. Using the FIELD_SEPARATOR stage property.
- C. Using the DELIMITER_CHAR table property.
- D. Using the COLUMN_SEPARATOR query parameter

ANSWER: A

Explanation:

Using the FIELD_DELIMITER file format option is correct because Snowflake parses delimited text files according to the file-format settings supplied to the COPY INTO command. For CSV data, FIELD_DELIMITER defines the character or character sequence that separates one field from the next. It can be specified inline through the command's FILE_FORMAT clause, for example FILE_FORMAT = (TYPE = CSV FIELD_DELIMITER = '|'), or defined once in a named file format and then referenced during loading. This keeps parsing rules explicit and reusable across load operations. If no value is supplied, Snowflake's CSV default field delimiter is a comma, so setting this option is the appropriate way to load pipe-, tab-, semicolon-, or otherwise custom-delimited files. The setting applies during the parsing of staged source files before their values are inserted into the target table. Snowflake also supports related file-format settings, such as record delimiters, quote/enclosure handling, escape characters, and null representation, allowing the complete structure of a delimited source file to be described consistently. See the Snowflake documentation for [COPY INTO <table>](#) and the [CREATE FILE FORMAT](#) syntax and options.

QUESTION NO: 12

What are the key benefits of the Snowflake multi-cluster shared data architecture? (Select TWO).

- A. It provides enhanced data security features.
- B. It optimizes data loading for unstructured data.
- C. It allows for independent scaling of compute and storage.
- D. It stores data in a columnar format to improve performance.
- E. It offers near-unlimited concurrency and elasticity.
- F. It allows for independent scaling of compute and storage and offers near-unlimited concurrency and elasticity.

ANSWER: F

Explanation:

The combined benefit of independent compute-and-storage scaling and near-unlimited concurrency with elasticity is correct. Snowflake's architecture separates persistent cloud storage from compute resources, which are provided by virtual warehouses. Because these layers are independent, an organization can resize or start additional compute capacity to meet workload needs without moving or resizing the data storage layer. This lets teams tune performance and cost according to the requirements of distinct workloads. Snowflake also supports multi-cluster virtual warehouses, which can automatically start additional clusters as demand increases. Each cluster accesses the same centrally managed data, enabling concurrent users and queries to scale horizontally while reducing query queuing. This combination is the central value of Snowflake's multi-cluster shared-data design: shared, centrally stored data paired with isolated and elastic compute. Snowflake describes the separation of storage and compute in its [key concepts documentation](#) and documents automatic cluster scaling and concurrency handling in its [multi-cluster warehouse documentation](#).

QUESTION NO: 13

Which of the following are examples of system-defined roles in Snowflake? (Choose any 3 options)

- A. USERADMIN
- B. CUSTOM_ROLE
- C. ACCOUNTADMIN
- D. SECURITYADMIN

ANSWER: A C D

Explanation:

Snowflake provides a set of system-defined roles that implement its built-in role-based access control hierarchy. **USERADMIN**, **ACCOUNTADMIN**, and **SECURITYADMIN** are all predefined account roles supplied by Snowflake and available in every account. USERADMIN is intended for identity and role administration, including creating users and roles. SECURITYADMIN manages security-related access control activities, such as managing grants, and inherits the capabilities of USERADMIN. ACCOUNTADMIN is the top-level system role, combining the privileges of SECURITYADMIN and SYSADMIN and providing broad administrative control over the Snowflake account. These roles can be granted to users or to other roles as part of an organization's access-control design. Snowflake recommends using the least-privileged role appropriate for an administrative task, reserving ACCOUNTADMIN for a small number of tightly controlled administrative users. The built-in role hierarchy enables delegated administration while maintaining centralized governance. Snowflake documents these roles, their inheritance relationships, and their intended administrative responsibilities in its access-control documentation. See [Access control overview](#) and [System-defined roles](#).

QUESTION NO: 14

Before below options what you need for executing INSERT command in Snowflake worksheet?

- A. Active Warehouse
- B. Schema
- C. Database
- D. All of them

ANSWER: A

Explanation:

Active Warehouse is the required selection among the listed choices because an `INSERT` statement is a DML operation that requires virtual warehouse compute to execute. A warehouse provides the compute resources Snowflake uses to compile and run SQL statements; the warehouse must be running, or Snowflake must be able to resume it, when the statement is submitted. The executing role must also have the necessary privileges, including usage access to the warehouse and permission to insert into the target table.

A current database and schema are convenient session-context settings when the target table is referenced with an unqualified or partially qualified name. However, they are not intrinsically required for an `INSERT` when the table is identified using its fully qualified name, such as `MY_DB.MY_SCHEMA.MY_TABLE`. Snowflake SQL supports qualified object names for this purpose. Thus, based strictly on the choices provided, an active warehouse is the necessary execution resource. See Snowflake's guidance on [virtual warehouses](#) and the syntax for the [INSERT command](#).

QUESTION NO: 15

What are the benefits of using the Snowsight data loading interface? (Select TWO).

- A. It creates permanent file formats that can be used to load data in the future.
- B. It allows a user to insert the records of a supported file into a table.
- C. It will try to detect data types.
- D. It allows a user to optimize data loading into a table.
- E. It lets a user merge file rows into the table records.

ANSWER: B C

Explanation:

Snowsight's data loading interface provides a guided way to load data from supported files into Snowflake tables without requiring the user to write the loading commands manually. It allows a user to insert the records of a supported file into a table, including a table that is created during the workflow or an existing target table selected by the user. This makes the interface particularly useful for interactive, small-scale, and exploratory ingestion tasks.

It will also try to detect data types when a user loads a file and chooses to create a table. Snowsight examines the file contents to infer column names and appropriate data types, then presents the detected schema for review. The user can adjust the inferred definitions before creating the table and completing the load, retaining control over the resulting table structure.

These capabilities streamline the common workflow of selecting a local file, reviewing its inferred structure, choosing a destination table, and loading the file's records. Snowflake documents this guided process in its Snowsight loading documentation: [Loading data using Snowsight](#) and [Load data into Snowflake](#).

QUESTION NO: 16

What is the purpose of a role hierarchy in Snowflake?

- A. To define the sequence of SQL queries
- B. To organize roles and grant inherited privileges
- C. To manage network settings
- D. To store raw data

ANSWER: B

Explanation:

To organize roles and grant inherited privileges is correct because Snowflake's access-control model uses roles as the primary mechanism for assigning and managing privileges. A role hierarchy is created by granting one role to another role. The receiving, or parent, role inherits the privileges granted to the child role, including the privileges that child role inherits from roles beneath it. When a user is granted the parent role, the user can use the permissions available through that hierarchy. This lets administrators define reusable functional roles, such as roles for read-only analytics or data engineering, and then combine them into broader departmental or administrative roles rather than repeatedly granting individual object privileges to every user. The result is a more scalable, centralized, and maintainable authorization design, particularly when people change teams or when access requirements evolve. Snowflake supports hierarchical role structures for both account roles and database roles, subject to its role-granting rules. Administrators can inspect these relationships and determine effective access by reviewing role grants. Snowflake recommends role-based access control patterns that use role hierarchies to simplify privilege administration and help implement least-privilege access. See Snowflake's [access control overview](#) and its documentation on [role hierarchy and privilege inheritance](#).

QUESTION NO: 17

What type of mode available in PARSE_DOCUMENT function? (Select two)

- A. CONTENT
- B. LAYOUT
- C. OMR
- D. OCR

ANSWER: B D

Explanation:

`LAYOUT` and `OCR` are the available processing modes for Snowflake's `PARSE_DOCUMENT` function. This function is used with a document staged in Snowflake and returns parsed document content as a structured `VARIANT` result, enabling document text to be used in SQL-based workflows and AI applications.

`OCR` is the optical-character-recognition processing mode. It extracts text from supported document formats, making document text available even when the source is image-based or scanned. This mode is appropriate when the primary requirement is to obtain the textual content of a document.

`LAYOUT` performs layout-aware parsing. In addition to recognizing text, it preserves information about the document's organization, such as elements, sections, and table-related structure. The richer structural output is useful when subsequent processing needs the relationships and placement of content, such as document retrieval, chunking, summarization, or extracting information from structured business documents.

Snowflake documents these values as the accepted `mode` argument values for `PARSE_DOCUMENT`. See the [PARSE_DOCUMENT function reference](#) and Snowflake's [Document AI parsing documentation](#) for supported modes and output behavior.

QUESTION NO: 18

Which of the following is true about data listings in the Snowflake Marketplace?

- A. They always require payment for access.
- B. They are limited to static data snapshots.
- C. They can be free or paid, depending on the provider 's terms.
- D. They are only accessible to Snowflake administrators.

ANSWER: C

Explanation:

They can be free or paid, depending on the provider 's terms. Snowflake Marketplace providers can publish listings that consumers access at no cost, or they can monetize listings through paid offerings. For paid listings, the provider defines the commercial terms and pricing model, while Snowflake facilitates secure access to the shared data product and the associated fulfillment workflow. This enables providers to distribute valuable data sets, applications, and other data products commercially without requiring consumers to build separate ingestion pipelines or maintain copied versions of the provider's data. A consumer obtains the listing through Snowflake, and access is governed by the listing's terms and the privileges granted through Snowflake's sharing mechanisms. Snowflake's Marketplace documentation describes both free and paid listings and explains that providers can create paid listings through the Provider Studio workflow. This commercial flexibility is a central Marketplace capability: a provider can make a listing broadly available for discovery while choosing the terms under which eligible consumers can obtain and use it. See Snowflake's [About listings](#) documentation and its guide to [creating paid listings](#).