Download ServiceNow CIS-DF Exam Dumps Free

Shared by Taylor on 17-06-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

A Data Center Manager is working with the CMDB CI Class Manager to define the relationship between Application Servers and the Applications they host. The company has multiple Application Servers that host one or more Applications.

Which describes the relationship between the Application Server table [cmdb_ci_app_server] and the Application table [cmdb_ci_appl]?

Options:

A- Many-to-many
B- One-to-many
C- Many-to-one
D- One-to-one

Answer:

A

Explanation:

In Data Foundations, ''Configuration'' includes modeling CI relationships in a way that reflects real-world technology dependencies and supports operational use cases such as impact analysis, troubleshooting, and service mapping. For ''Application Servers host Applications,'' the scenario describes that multiple Application Servers can host one or more Applications. In most enterprise environments, the reverse is also true: an Application can be hosted across multiple Application Servers (for example, horizontal scaling, clustered deployments, active-active architectures, and separate servers for different tiers or components of the same application).

Because both sides can have multiple related records, the correct logical relationship is many-to-many:

One Application Server hosts many Applications

One Application can be hosted on many Application Servers

In ServiceNow CMDB terms, many-to-many relationships are represented through the CMDB relationship model (parent/child relationships), enabling the platform to store multiple relationship records linking servers and applications without forcing an unnatural single-reference constraint. This supports better CMDB quality and stronger downstream outcomes---especially for incident triage (''what apps are affected if this server is down?'') and change impact analysis

(''what servers could be impacted if we change this app component?'').

A one-to-many or many-to-one model would incorrectly restrict either the server to one application or the application to one server, which does not align with common hosting patterns and would reduce the CMDB's usefulness and accuracy.

Question 2

Question Type: MultipleChoice

A customer wants recently imported server records to be automatically reclassified into more specific CMDB classes after being discovered by ServiceNow Discovery.

During the discovery process, if existing Server records are reclassified into the Linux Server and Windows Server classes, which reclassification operation occurs?

Options:

A- Class Switch
B- Class Downgrade
C- Class Upgrade

Answer:

C

Explanation:

In the CMDB class hierarchy, Server is a generic parent class, while Linux Server and Windows Server are more specific child classes. When ServiceNow Discovery detects sufficient evidence (such as OS signatures) to move a CI from a generic class to a more specialized one, this action is called a Class Upgrade.

A Class Upgrade occurs when a CI is reclassified down the hierarchy into a more specific subclass, enriching the record with additional attributes, behaviors, and discovery patterns appropriate to that class. This is a standard and expected behavior in mature CMDB implementations and aligns with Data Foundations best practices.

A Class Switch would imply lateral movement between unrelated classes, which is not what happens here. A Class Downgrade would move a CI from a specific class back to a more generic one, typically when discovery confidence is reduced---not the case in this scenario.

By performing class upgrades automatically, Discovery improves CMDB accuracy, reporting precision, and service mapping quality without manual intervention.

Therefore, the correct answer is C -- Class Upgrade.

Question 3

Question Type: MultipleChoice

A Data Center Manager is working with the CMDB CI Class Manager to define the relationship between Application Servers and the Applications they host. The company has multiple Application Servers that host one or more Applications.

Which describes the relationship between the Application Server table and the Application table?

Options:

A- Many-to-many
B- Many-to-one
C- One-to-one
D- One-to-many

Answer:

A

Explanation:

In CMDB modeling, accurately defining relationships is critical for impact analysis, service mapping, and Change Management. In this scenario, Application Servers can host multiple Applications, and Applications can also run across multiple Application Servers (for example, in clustered, load-balanced, or distributed architectures).

This architectural reality defines a many-to-many relationship between the Application Server table and the Application table.

In ServiceNow, many-to-many relationships are common for application hosting models, especially in modern environments that use horizontal scaling, redundancy, or containerized workloads. Modeling this correctly ensures that incidents, changes, and outages affecting a single server can be accurately traced to all impacted applications---and vice versa.

A one-to-many or many-to-one relationship would incorrectly assume exclusivity in one direction,

which does not reflect real-world application deployment patterns. A one-to-one relationship would be even more restrictive and inaccurate.

Therefore, the correct relationship type is A -- Many-to-many, which aligns with CMDB best practices and CSDM service modeling principles.

Question 4

Question Type: MultipleChoice

(Choose 2 options)

A CMDB Administrator wants to create a CMDB query to find all databases located in Seattle that are connected to application services. They also want to include incidents related to those databases.

Which actions should be taken to build this query?

Options:

A- Add to the canvas the Incident table from the Non-CMDB Tables list
B- Add property columns to the Application Service node
C- Add a filter to the Database node for Location = Seattle
D- Set the relationship level to up to 2nd-level relationships

Answer:

A, C

Explanation:

When building advanced CMDB queries using CMDB Query Builder in ServiceNow, the correct approach is to model CI scope, relationships, and task context explicitly on the canvas.

To limit results to databases in a specific location, the administrator must filter the Database CI node by the Location attribute. Therefore, Option C is required to scope the query to Database CIs where Location = Seattle.

To include Incidents related to those databases, the Incident table must be added from the Non-CMDB Tables list and linked through the task_ci relationship. This is exactly what Option A provides. CMDB Query Builder separates CMDB tables (CIs) from task and transactional tables, so incidents must be explicitly added from the Non-CMDB section.

Option B is incorrect because property columns on Application Services do not scope databases or incidents. Option D is unnecessary because relationship depth alone does not include non-CMDB task data and does not filter by location.

Thus, the correct actions are A (add Incident table) and C (filter Database by location).

Question 5

Question Type: MultipleChoice

Where can a CMDB 360 / Multisource CMDB Saved Query be viewed and created in the CMDB Workspace?

Options:

A- Coverage window on the CMDB 360 tab
B- Saved queries window on the Insights tab
C- CMDB Query Builder
D- Saved queries window on the CMDB 360 tab

Answer:

D

Explanation:

In Data Foundations, ''Insight'' focuses on turning configuration data into actionable visibility. CMDB 360 (often associated with Multisource CMDB capabilities) provides a guided experience in CMDB Workspace for exploring CI data and running queries across sources. Within this experience, Saved Queries are managed directly under the CMDB 360 area of CMDB Workspace because they are part of the CMDB 360 query-and-analysis workflow.

A CMDB 360 Saved Query is not the same as a generic CMDB Query Builder query created from classic navigation modules. The CMDB 360 experience typically uses a tile/window approach where users can view existing saved queries, create new queries, modify them, and run them to retrieve CMDB 360 data aligned to multisource reporting and analysis. Keeping saved queries on the CMDB 360 tab makes them easy to discover and reuse for repeatable insights (for example, coverage and source comparisons, CI data quality checks by source, and targeted investigation of records).

The Insights tab in CMDB Workspace is generally used to understand adoption and health-related insights about CMDB features, not as the primary location for creating CMDB 360 multisource

saved queries. ''Coverage'' is a specific lens/view and does not represent the saved-query creation workspace. ''CMDB Query Builder'' is a related capability, but the question explicitly asks where CMDB 360/Multisource Saved Queries are viewed and created in CMDB Workspace, which is the Saved queries window on the CMDB 360 tab.

Question 6

Question Type: MultipleChoice

A CMDB Manager wants to improve data quality using the CMDB Health Dashboard.

What needs to happen to generate CMDB health scores?

Options:

A- The scheduled jobs for the CMDB Health Dashboard must be activated
B- Nothing, CMDB health scores are calculated by default
C- The plugin, CMDB health calculation, needs to be installed

Answer:

A

Explanation:

In ServiceNow, the CMDB Health Dashboard does not calculate health scores in real time by default. Instead, health scores are generated and refreshed by scheduled calculation jobs that evaluate CI data against defined health rules across the dimensions of completeness, correctness, and compliance.

To generate and maintain CMDB health scores, the scheduled jobs for CMDB Health must be active. These jobs periodically scan the CMDB, apply health rules (for example, required attributes populated, lifecycle status compliance, certification results), and calculate scores that are displayed on the dashboard and scorecards. Without these scheduled jobs running, the dashboard cannot produce current or meaningful health metrics.

Option B is incorrect because CMDB health scoring is not automatic or real-time; it depends on scheduled processing. Option C is also incorrect because CMDB Health is part of the core CMDB/Data Foundations capability in ServiceNow and does not require a separate ''CMDB health calculation'' plugin to be installed in modern implementations.

Activating and maintaining these scheduled jobs ensures that health scores remain accurate,

trendable over time, and useful for governance decisions. This is a foundational requirement for using the CMDB Health Dashboard as a data quality improvement tool.

Therefore, the correct answer is A -- The scheduled jobs for the CMDB Health Dashboard must be activated.

Question 7

Question Type: MultipleChoice

What is the difference between Data Certification and Attestation policies when managing a CI?

Options:

A- Attestation requires correcting specific attributes of a CI, while Data Certification tracks acknowledgement the CI still exists
B- Attestation can be scheduled, while Data Certification cannot be scheduled
C- Attestation can be assigned to a group or an individual, while Data Certification can only be assigned to an individual
D- Attestation tracks acknowledgement the CI still exists, while Data Certification requires validating specific attributes of a CI

Answer:

D

Explanation:

Within ServiceNow CMDB governance, Attestation and Data Certification serve distinct but complementary purposes. The key difference lies in what is being validated.

Attestation is focused on existence and ownership confirmation. When a CI is attested, the assigned user or group is asked to confirm that the CI still exists, is still relevant, and is still owned or managed by the appropriate team. No detailed attribute-level validation is required. This lightweight process is commonly used to prevent ''ghost CIs'' from lingering in the CMDB.

Data Certification, on the other hand, is more rigorous. It requires the certifier to validate specific attributes of the CI, such as lifecycle status, support group, environment, or service relationships. Certification ensures data correctness and completeness, which directly impacts CMDB Health scores and downstream processes like Change and Incident Management.

Options A, B, and C incorrectly describe these mechanisms or their assignment and scheduling

capabilities. Both attestation and certification can be scheduled and assigned flexibly, but their validation depth is what truly differentiates them.

Therefore, Option D correctly describes the distinction: attestation confirms existence, while data certification validates CI attributes.

Question 8

Question Type: MultipleChoice

A CMDB Administrator needs to import external data into the CMDB. To reduce the risk of creating duplicates and prevent updates from unauthorized sources, it must be ensured that the Identification and Reconciliation Engine (IRE) is not bypassed.

What is the recommended method to import data into the CMDB utilizing the Identification and Reconciliation API?

Options:

A- IntegrationHub ETL
B- Table API (REST API or SOAP API)
C- Import Sets and Transform Maps

Answer:

A

Explanation:

In ServiceNow, protecting CMDB data quality during ingestion is a core Data Foundations principle. The Identification and Reconciliation Engine (IRE) is designed to ensure that CI records are uniquely identified, merged correctly, and protected from unauthorized overwrites. Any ingestion method that bypasses IRE introduces a high risk of duplicates and data corruption.

IntegrationHub ETL is the recommended method because it is natively designed to work with the Identification and Reconciliation API. When properly configured, IntegrationHub ETL ensures that incoming data is processed through IRE, applying identification rules, reconciliation rules, and source precedence. This allows multiple data sources to coexist safely while maintaining CMDB integrity.

Option B (Table API) is explicitly discouraged for CMDB ingestion because it writes directly to CMDB tables and bypasses IRE entirely, making it one of the most common causes of duplicate

and conflicting CI records. While REST and SOAP APIs are powerful, they are not safe for CMDB ingestion unless they explicitly invoke the IRE API, which most generic table integrations do not.

Option C (Import Sets and Transform Maps) can be configured to call IRE, but this requires additional scripting and strict governance. Because of this complexity and higher risk of misconfiguration, it is not the recommended approach when safer, purpose-built options exist.

Therefore, IntegrationHub ETL is the verified and best-practice answer, making Option A correct.

Question 9

Question Type: MultipleChoice

(Choose 2 options)

Which ServiceNow solutions create automatic relationships?

Options:

A- IntegrationHub ETL
B- Service Mapping
C- Discovery
D- Workflow Studio

Answer:

B, C

Explanation:

Automatic relationship creation is fundamental to maintaining a service-aware and trustworthy CMDB. In ServiceNow, this capability is primarily delivered by Discovery and Service Mapping.

Discovery (Option C) automatically identifies infrastructure components---such as servers, network devices, and storage---and creates technical relationships between them (for example, ''runs on,'' ''connected to,'' or ''depends on''). These relationships form the backbone of infrastructure dependency mapping.

Service Mapping (Option B) builds on Discovery by creating application- and service-level relationships. It maps how application components interact across servers, databases, and middleware, resulting in accurate Application Service models aligned with CSDM. These relationships are created and maintained automatically as the environment changes.

Option A (IntegrationHub ETL) focuses on data ingestion and transformation; it does not inherently create or maintain relationships unless explicitly scripted. Option D (Workflow Studio) orchestrates processes and automations but does not discover or infer CI relationships.

Therefore, the ServiceNow solutions that create automatic relationships are Service Mapping and Discovery, making Options B and C correct.

Question 10

Question Type: MultipleChoice

A CMDB Administrator wants to improve data quality related to the CSDM.

Which action should the Administrator take to meet this goal?

Options:

A- Use the CSDM Data Foundations Dashboard
B- Start the ServiceNow Health Scan
C- Use the default configured CMDB Health Dashboard

Answer:

A

Explanation:

To specifically improve data quality related to CSDM, the most effective and prescribed action is to use the CSDM Data Foundations Dashboard. In ServiceNow, this dashboard is purpose-built to assess and improve CSDM alignment, not just general CMDB hygiene.

The CSDM Data Foundations Dashboard focuses on service modeling readiness, highlighting gaps such as missing service ownership, incomplete relationships between Business Applications and Application Services, unmanaged services, and misaligned lifecycle states. It provides Get Well Playbooks that guide administrators through structured remediation using Analyze Data, Fix Data, and Govern Data plays---directly tied to CSDM outcomes.

Option C (default CMDB Health Dashboard) is valuable, but it measures generic CMDB data quality dimensions (completeness, correctness, compliance) and does not specifically evaluate CSDM constructs or service modeling maturity. Option B (ServiceNow Health Scan) provides platform-wide configuration and performance recommendations, but it is not focused on CMDB or CSDM data quality.

Therefore, to improve CSDM-specific data quality, the administrator should use the CSDM Data Foundations Dashboard, making Option A the correct answer.

Question 11

Question Type: MultipleChoice

A Service Owner is using Unified Map to understand the composition of a service but wants to filter irrelevant information.

Which options are available to the Service Owner from the filter panel? (Choose 2 options)

Options:

A- CI type
B- Discovery source
C- Managed by group
D- Business criticality

Answer:

A, D

Explanation:

The Unified Map in ServiceNow provides a consolidated view of services and their underlying components, integrating Discovery and Service Mapping data. To make this view actionable, Service Owners can apply filters to focus on relevant elements and reduce visual noise.

Filtering by CI type (Option A) is a core capability. It allows the Service Owner to show or hide categories such as servers, databases, load balancers, or applications---making it easier to analyze specific layers of the service.

Filtering by Business Criticality (Option D) is also available and highly valuable. This enables Service Owners to prioritize views around high-impact components, ensuring attention is focused on CIs that pose the greatest risk to service delivery.

Option B (Discovery source) is not typically exposed as a Unified Map filter because the map focuses on operational and service context, not ingestion provenance. Option C (Managed by group) is a governance attribute and is not a standard visual filter within the Unified Map.

Thus, the correct filter options are A -- CI type and D -- Business criticality.