



Certified Implementation Specialist Data Foundations (CMDB and CSDM)

ServiceNow CIS-DF

Version Demo

Total Demo Questions: 25

Total Premium Questions: 252

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

PASSQUEEN.COM

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Data Foundations	24
Topic 2, Configuration Management Database (CMDB)	173
Topic 3, Common Service Data Model (CSDM)	53
Topic 4, Mix Questions	2
Total	252

QUESTION NO: 1

A CMDB Administrator is evaluating solutions to automate the recurring ingestion of third-party infrastructure data into the CMDB.

Which capabilities does ServiceNow IntegrationHub ETL provide?

(Choose two options.)

- A. Provides a guided interface to extract data from external sources, transform it to match ServiceNow data structures, and load it into target tables.
- B. Requires knowledge of SQL and scripting languages to configure data transformations.
- C. Replaces Discovery and Service Mapping by continuously monitoring all infrastructure components in real time.
- D. Natively integrates with the Identification and Reconciliation Engine to apply identification and reconciliation rules during CMDB data ingestion.

ANSWER: A D

Explanation:

IntegrationHub ETL provides a guided, low-code process for bringing external data into ServiceNow on a repeatable basis. It supports defining a source connection, extracting source data, mapping fields and relationships to the appropriate ServiceNow target structures, applying transformations, and loading the resulting records into target tables. This makes it appropriate for scheduled or recurring ingestion of data supplied by third-party infrastructure-management tools and other external systems.

For CMDB ingestion, IntegrationHub ETL works with the Identification and Reconciliation Engine. Identification rules determine whether incoming configuration-item data matches an existing record or requires a new record, while reconciliation rules and source precedence govern whether a source can update particular attributes. This allows imported data to be processed according to the organization's CMDB identification and data-governance policies rather than simply creating duplicate or uncontrolled updates.

These capabilities support governed integration of externally supplied data while retaining CMDB data-quality controls. See ServiceNow's [IntegrationHub ETL documentation](#) and its [Identification and Reconciliation Engine documentation](#).

QUESTION NO: 2 - (HOTSPOT)

A CMDB Owner starts the CSDM journey and needs to become familiar with the CSDM domains.

Drag the CMDB objects to the correct CSDM domains.

The diagram shows a drag-and-drop interface. On the left, there are four boxes representing CMDB objects: 'Application Service', 'Business Application', 'Business Process', and 'Business Service'. In the center, there is an 'Answer Area' with four empty dashed boxes. On the right, there are four labels representing CSDM domains: 'Design and Planning domain', 'Foundation domain', 'Service Delivery domain', and 'Sell / Consume domain'.

ANSWER:

Explanation:

The mapping shown correctly places each CMDB object in the CSDM domain that reflects its role in planning, foundational data management, operational delivery, or consumer-facing service consumption. A Business Process belongs in the Design and Planning domain because it describes how an organization performs work and provides the planning context used to design supporting services and applications. A Business Application is placed in the Foundation domain because it is a core enterprise record used for application ownership, lifecycle, portfolio context, and relationships across the CMDB. An Application Service belongs in the Service Delivery domain because it represents an operationally managed service that supports service operations, including visibility into service health, incidents, changes, and dependencies. A Business Service belongs in the Sell / Consume domain because it represents a service that is understood and consumed from a business or customer perspective. This organization helps keep business planning concepts, foundational reference data, operational service records, and consumer-oriented services clearly separated while preserving the relationships needed for reliable service management. ServiceNow describes CSDM as the common framework for organizing service-related data and establishing consistent relationships across products and processes. See the [ServiceNow CSDM overview](#) for further guidance on using the model to establish a scalable service data foundation.

QUESTION NO: 3

A production web portal is represented in the CMDB by a deployed service record with relationships to its web servers, application servers, and database. Incident and Change Management teams need to use this record for operational impact analysis.

Which CSDM class should represent the deployed web portal?

- A. Application Service
- B. Business Application
- C. Business Service
- D. Service Offering

ANSWER: A

Explanation:

Application Service is correct because it represents a deployed and operational application service together with the configuration items and dependencies that support it. It provides the technical service context required for incident triage, impact analysis, service mapping, and change-risk assessment.

A Business Application is a logical portfolio record used for planning and application architecture. A Business Service and its Service Offerings represent business-facing services and consumable variations rather than the deployed technical topology of the portal.

QUESTION NO: 4

An organization wants to distinguish its application portfolio records from the deployed services that support incident, change, and operational-impact processes.

Which statements correctly apply the CSDM classes? (Choose two options.)

- A. A Business Application represents an application from a portfolio and planning perspective
- B. An Application Service represents a deployed, operational application service
- C. A Service Offering represents the complete server topology for an application
- D. A Business Service represents a single deployed database instance

ANSWER: A B

Explanation:

A Business Application represents an application from a portfolio and planning perspective is correct. It provides the logical application context used for business and architecture decisions, including ownership, lifecycle, and portfolio management.

An Application Service represents a deployed, operational application service is also correct. Application Services are used to model the operational service and its supporting dependencies, enabling service-aware incident handling, change assessment, and impact analysis. A Service Offering represents a consumable service variation, not a server topology, and a Business Service is not used to represent a single deployed database instance.

QUESTION NO: 5

(Choose 2 options)

A CMDB Administrator needs to create a new CI class for an Internet of Things (IoT) Sensor in ServiceNow.

What are the recommended practices for this activity?

- A. Delete an unused class and replace it with the new one
- B. Install or update the CMDB CI Class Models Store application and verify the class does not already exist
- C. Add a new class under an appropriate parent class
- D. Modify an existing class

ANSWER: B C

Explanation:

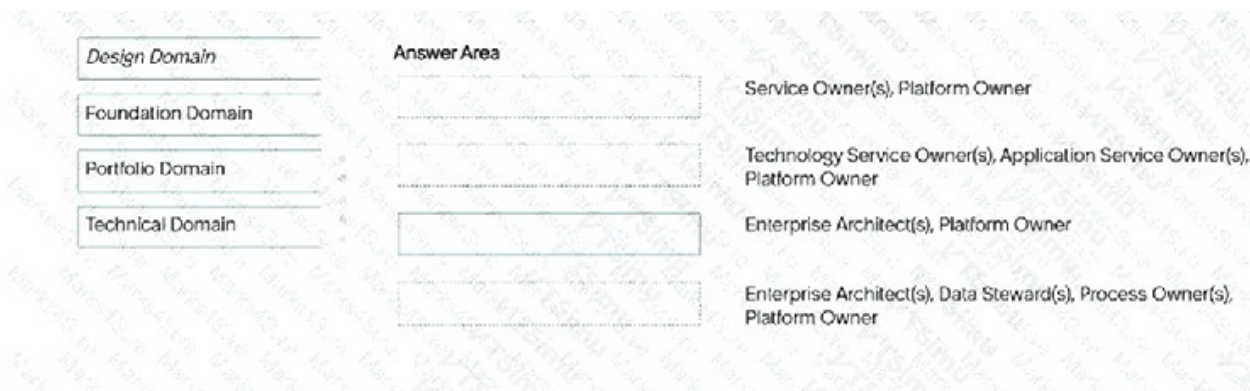
Before introducing an Internet of Things sensor class, the administrator should install or update the CMDB CI Class Models Store application and verify the class does not already exist. ServiceNow's CMDB class-model content helps customers use standardized, supported class definitions and assess whether delivered data models already meet the requirement. Reviewing the available class models first prevents unnecessary duplication, promotes consistent identification and relationship modeling, and reduces governance overhead as the CMDB evolves.

When a distinct class is justified, adding a new class under an appropriate parent class is the recommended extensibility approach. ServiceNow CMDB classes are organized in an inheritance hierarchy, so selecting the closest suitable parent enables the sensor class to inherit applicable common attributes and platform behavior. This produces a more coherent model, limits needless custom fields, and makes the resulting CI type easier to govern, populate, and use in downstream processes. For an IoT sensor, the chosen parent should reflect its actual managed-object characteristics and intended operational use. These practices align with ServiceNow guidance to use the out-of-box CMDB data model wherever possible and extend it carefully only for validated requirements. See the [ServiceNow CMDB documentation](#) and the [CMDB Class Manager documentation](#).

QUESTION NO: 6 - (SIMULATION)

A Platform Owner is building the governance team to support CSDM.

Drag the domain to the roles that make up the governance team.



ANSWER: See the explanation for the answer

Explanation:

Drag the governance-team role group to each CSDM domain as follows:

This assigns accountability based on the data governed in each domain: design is governed by architecture, data, and process stakeholders; foundation is platform and architecture reference data; portfolio is owned by service owners; and technical service/application data is owned by the respective technology and application service owners.

QUESTION NO: 7

A Manager needs information on how to correctly establish relationships between Infrastructure CIs, Technology Management Offerings, Technical Service Offerings, and Application Services within the CMD

B.

Which CSDM domain would provide this information?

- A. Build and Integration
- B. Service Consumption
- C. Foundation
- D. Design and Planning
- E. Service Delivery

ANSWER: E

Explanation:

Service Delivery is the CSDM domain that provides the relationship guidance needed to connect the operational components that support delivered services. It describes how infrastructure configuration items are associated with application services and with the technical-service and technical-service-offering records used to represent the technology capabilities being delivered. These relationships create the service context needed for operational practices such as impact analysis, incident and change management, service health, and reporting. In practical terms, the domain helps an organization model the path from an application service to its supporting technology and infrastructure, while also identifying the applicable technical offerings through which that capability is delivered. Establishing these links consistently is essential because CSDM is not only a collection of tables; it is a prescriptive model for representing service-related data and its dependencies in the CMDB. ServiceNow's CSDM guidance organizes the CMDB model into domains so teams can determine where specific service and CI relationship patterns belong. See the [ServiceNow CSDM overview](#) and the [ServiceNow CSDM community resources](#) for CSDM modeling guidance.

QUESTION NO: 8

CMDB class owners are receiving tasks under the “My Work” tab in the CMDB Workspace.

Which CMDB management tool is generating those tasks?

- A. De-duplication templates
- B. CMDB Data Manager
- C. CMDB Health Dashboard

ANSWER: B

Explanation:

CMDB Data Manager is the CMDB management tool that generates these actionable tasks. It supports ongoing CMDB data lifecycle governance by applying Data Manager policies to configuration items and creating work for the appropriate accountable users when an item requires review or action. This operationalizes ownership of CMDB data rather than merely reporting on its condition. Class owners can receive the resulting assignments in the CMDB Workspace *My Work* experience, where they can review and complete the actions associated with their owned CMDB classes and configuration items.

Data Manager policies can be used to manage lifecycle activities such as attestation, archival, and deletion, based on criteria and schedules defined by the organization. When policy processing identifies CIs that meet the policy conditions, CMDB Data Manager produces tasks so that the responsible owner or steward can make an informed governance decision. This provides a controlled, auditable mechanism for keeping CMDB records current throughout their lifecycle and assigning accountability directly to the people responsible for the relevant data domain.

ServiceNow documents CMDB Data Manager and its policy-driven lifecycle capabilities in the [CMDB Data Manager documentation](#) and describes workspace-based CMDB operations in the [CMDB Workspace documentation](#).

QUESTION NO: 9

A CMDB Administrator has configured foundation group data for CIs belonging to the Linux Server class through two different interfaces. CI Class Manager specifies **Linux Support** as the Managed by Group, while a Technology Management Service Offering specifies **Enterprise IT Services** as the Managed by Group and **Change Management Team** as the Change Group for the same CI population.

When Linux Server CIs are discovered and ingested into the CMDB, which group will be assigned as the Managed by Group for these instances?

- A. Enterprise IT Services
- B. Linux Support
- C. No group, because the conflicting assignments cause a data-validation error.
- D. Change Management Team

ANSWER: A

Explanation:

Enterprise IT Services is the Managed by Group assigned to the discovered Linux Server CIs. A Technology Management Service Offering can be associated with a CI population through its Dynamic CI Group. When the offering is configured with operational group information, ServiceNow synchronizes the applicable offering group values to the CIs that are members of that population. This enables the ownership and operational-responsibility data on the service offering to remain aligned with the CIs that deliver it.

In this scenario, the offering’s Managed by Group value is Enterprise IT Services, so that is the value applied to the Managed by Group field for the matching Linux Server instances. The separately configured Change Group remains a distinct operational group attribute; it does not replace the offering’s Managed by Group value. This behavior supports CSDM’s objective of maintaining consistent accountability data between technology service offerings and their supporting configuration items, including as CIs are added through Discovery or other CMDB ingestion processes.

QUESTION NO: 10

The CMDB Administrator has been asked to establish Configuration Management with a functional CMD

B.

Which factor is most critical for successfully operationalizing the CMDB in ServiceNow?

- A. Establishing clear governance and continuously monitoring CMDB health
- B. Relying on automated discovery tools to maintain and update CMDB records
- C. Allowing IT teams to modify CMDB records as needed to promote flexibility in data management
- D. Populating the CMDB with as much data as possible to ensure a comprehensive inventory of CIs

ANSWER: A

Explanation:

Establishing clear governance and continuously monitoring CMDB health is most critical because a functional CMDB is an ongoing operational capability, not simply a repository populated once. Governance establishes accountable roles, decision rights, standards, CI ownership, lifecycle expectations, and processes for maintaining CI and relationship data. These controls ensure that the CMDB supports trusted operational decisions across practices such as incident, problem, change, and service management.

Continuous CMDB health monitoring is the practical mechanism for validating that this governance is working. ServiceNow CMDB Health evaluates quality dimensions including completeness, correctness, and compliance, enabling CMDB owners to identify gaps, prioritize remediation, and sustain confidence in the data over time. This focus aligns with the CSDM principle that CMDB data must be purposeful, governed, and maintained to support defined business and IT outcomes. A governed health-management approach also accommodates multiple data sources and ongoing infrastructure change while preserving accountability for the resulting records and relationships.

For further guidance, see ServiceNow's [CMDB Health documentation](#) and the [Common Service Data Model documentation](#).

QUESTION NO: 11

A company wants to track regulatory compliance. ServiceNow has an artifact type called an information object as part of the CSDM framework.

What is the purpose of an information object?

- A. It describes data in general on a group of Configuration Items.
- B. It describes data exchanged between an API interface and an Application.
- C. It describes the logical data to the Business Applications.

ANSWER: C

Explanation:

An information object describes the logical data associated with business applications. It supplies a business-level representation of the information an application processes, stores, transmits, or otherwise uses, rather than representing a specific database table, file, server, or interface implementation. This distinction makes information objects particularly valuable for regulatory compliance, privacy, risk, and audit reporting: an organization can identify business applications that handle categories of sensitive or regulated information, such as personally identifiable information, payment information, health information, or financial records.

Within CSDM, relating an information object to the relevant business application establishes meaningful context between an application and the data it supports. That context enables stakeholders to assess data handling obligations and application impacts in business terms, while technical implementation details can remain modeled separately through the appropriate application-service and configuration-item relationships. The model therefore supports a consistent, reusable data classification approach across the organization, even where multiple applications use the same type of logical information. ServiceNow's CSDM guidance describes the model and its business-application context in the [CSDM product model documentation](#); see also the [ServiceNow CSDM community resources](#).

QUESTION NO: 12

A Configuration Manager is managing a CI class in the CMDB. The identification rules require an update.

Where can the Configuration Manager view and configure the existing identification rules for the class?

(Choose two options.)

- A. IRE Application
- B. CI Class Manager
- C. API Integrations
- D. CI Identifiers module

ANSWER: B D

Explanation:

CI Class Manager is a valid place to view and maintain identification rules in the context of a specific CMDB class. It provides a class-focused management experience, bringing together class details and CMDB governance configuration. From the applicable class, a Configuration Manager can examine the identification-rule configuration associated with that class and make the required changes while retaining the broader context of the class hierarchy and attributes.

The **CI Identifiers module** is also correct because it provides direct access to the CI identifier records used by the Identification and Reconciliation Engine. An identifier defines the identifying attributes and the ordered identifier entries that IRE uses to determine whether inbound data matches an existing configuration item or should result in creation of a new one. This module therefore supports reviewing and editing the underlying identification-rule definitions for CMDB classes.

Using both locations supports class-centric administration and direct identifier-rule administration. ServiceNow documents identification rules as part of IRE configuration and describes CI Class Manager as the interface for managing CMDB class configuration. See the [ServiceNow Identification and Reconciliation Engine documentation](#) and [CI Class Manager documentation](#).

QUESTION NO: 13

A CMDB Data Manager needs to access the ServiceNow platform to create, publish, and manage policies that automate and govern CI lifecycle operations, ensuring the CMDB remains healthy and efficient.

Where can the Data Manager do this?

- A. CI Class Manager
- B. CMDB Workspace – Management tab
- C. Service Operations Workspace
- D. CMDB Workspace – CMDB 360 tab

ANSWER: B

Explanation:

CMDB Workspace – Management tab is the correct location because it provides the CMDB Data Manager experience used to administer lifecycle-management policies for configuration items. Data Manager policies establish repeatable, governed handling for CIs at different stages of their lifecycle, including activities such as attestation, archival, and deletion. A Data Manager can define the policy conditions and actions, review policy details, publish policies when they are ready for use, and monitor or maintain the policies over time. This gives CMDB owners an operational control point for applying lifecycle governance consistently rather than depending on manual cleanup activity. Automating these processes helps reduce stale, unnecessary, or unmanaged CI records and supports a healthier, more efficient CMDB. ServiceNow documents CMDB Data Manager as a CMDB capability for managing CI lifecycle operations through policies, and places this work in the Management area of CMDB Workspace. For further details, see the [ServiceNow CMDB Data Manager documentation](#) and the [ServiceNow CMDB product overview](#).

QUESTION NO: 14 - (SIMULATION)

A new ServiceNow customer is assembling a Configuration Management team to support their CMD

B.

Drag each role to its corresponding job description.

	Answer Area	
CI Analyst		Has read-only access to CMDB data and to basic user interface such as CMDB reports and dashboards
CMDB Process Owner		Accountable for managing all elements that make up a portfolio throughout their entire lifecycle
Configuration Manager/CMDB Admin		Manages assigned CI tables and keeps records updated and resolves tasks related to CMDB records
Service or Product Owner		Obtains highest level role for CMDB privileges

ANSWER: See the explanation for the answer

Explanation:

Match the roles to the descriptions as follows:

QUESTION NO: 15

A CMDB Administrator reviews the CMDB Data Foundations Dashboard and finds several Critical and High priority results. The Administrator wants to improve data quality without masking valid findings.

Which actions should the Administrator take? (Choose two options.)

- A. Prioritize Critical and High priority metrics
- B. Use the remediation playbook associated with each metric
- C. Create exclusion rules for the affected CIs to improve the metric score
- D. Delete each CI included in a low-scoring metric

ANSWER: A B

Explanation:

Prioritize Critical and High priority metrics is correct because these findings represent the most significant data-foundation gaps and provide a risk-based starting point for remediation. **Use the remediation playbook associated with**

each metric is also correct because the playbook provides guided investigation, root-cause information, and actions for correcting the underlying data or governance issue.

Excluding records simply to raise a score suppresses the signal without resolving the condition. Deleting CIs solely because they appear in a dashboard result can remove operationally relevant data and is not an appropriate general remediation action.

QUESTION NO: 16

A Windows administration team needs to organize Windows Server CIs for targeted Incident Management and Change Management workflows.

Which approaches can the team use to create logical groupings within CMDB groups?

(Choose two options.)

- A. Manually select specific Windows Server instances and assign them to a named CMDB group.
- B. Use tag-based queries to automatically assign Windows Server CIs to a named CMDB group.
- C. Use encoded queries to dynamically filter and group Windows Server CIs based on specific attribute conditions.
- D. Use reconciliation rules to assign CIs to groups based on data-source priority.

ANSWER: A C

Explanation:

CMDB groups support organizing configuration items into useful operational populations, such as Windows Server CIs that need common incident-routing, change-planning, ownership, or reporting treatment. “Manually select specific Windows Server instances and assign them to a named CMDB group.” is correct because a CMDB group can have explicitly defined membership. This approach is appropriate when the team needs a curated, stable set of servers that is not solely determined by CI attribute values.

“Use encoded queries to dynamically filter and group Windows Server CIs based on specific attribute conditions.” is also correct because dynamic CMDB groups derive their membership from query conditions. The team can define conditions using attributes such as CI class, operational status, location, environment, assignment group, or other CMDB fields. Membership is then maintained according to whether each CI meets the defined criteria, which makes the group suitable for an evolving population of Windows servers.

Using both static membership and query-driven membership lets administrators choose the grouping model that fits the operational need: direct selection for intentional exceptions or fixed scopes, and dynamic filtering for repeatable, attribute-based scopes. ServiceNow documents CMDB groups as a mechanism for collecting CIs and supports query-based identification of CMDB records. See the [ServiceNow CMDB groups documentation](#) and the [CMDB Query Builder documentation](#).

QUESTION NO: 17

An organization is defining end-of-life handling for CIs that are no longer used to deliver active services. The organization needs a governed process that first changes the operational lifecycle state and can later reduce the active CMDB data population while retaining records according to policy.

Which CMDB Data Manager end-of-life policy types support this requirement? (Choose two options.)

- A. Archive
- B. Retire
- C. Disposed
- D. Lost

ANSWER: A B

Explanation:

Retire is correct because it manages the transition of eligible CIs out of active operational use. A retired CI remains available for historical context while clearly indicating that it is no longer an active component of service delivery.

Archive is also correct because it supports moving eligible records out of the active CMDB population according to retention requirements. Together, retirement and archival provide a controlled lifecycle approach. Disposed and Lost are asset-oriented conditions that do not represent CMDB Data Manager end-of-life policy types.

QUESTION NO: 18

The CMDB Configuration Management team has successfully developed a healthy and trusted CMDB. They have integrated discovered infrastructure data, accurately referenced non-discoverable data (such as change and support group information), and made the CMDB service-aware using Service Mapping.

How will these improvements enhance the change management process? (Choose Two)

- A. Ensures that no changes result in service downtime, regardless of planning or execution
- B. Provides insight into the potential impact of the change
- C. Automatically schedules and deploys changes without human review or approval
- D. Enables auto-population of the assignment group field to dynamically route changes

ANSWER: B D

Explanation:

“Provides insight into the potential impact of the change” is correct because a service-aware CMDB provides the dependency relationships needed to understand a change in business context. Service Mapping connects applications and business services to the infrastructure CIs that support them. When a change is proposed against a CI, Change Management can use those relationships to identify potentially affected services, applications, and dependent components. This produces more reliable impact assessments, supports informed risk evaluation, and helps change personnel communicate with the teams responsible for affected services.

“Enables auto-population of the assignment group field to dynamically route changes” is also correct because trusted CMDB data includes operational ownership information that discovery does not normally provide, such as support groups and assignment groups. When CIs and services are consistently related to the responsible groups, Change Management can use that CMDB context to populate or route work to the appropriate team. This reduces manual lookup effort and helps ensure that the people accountable for the affected service or infrastructure participate in planning and implementation. These capabilities depend on accurate CI records, governed ownership attributes, and service relationships—the core outcomes of a healthy CMDB and Service Mapping. See ServiceNow's [Service Mapping overview](#) and [Change Management overview](#).

QUESTION NO: 19

A CMDB Administrator finds duplicate network-device CIs created before the organization implemented consistent ingestion controls. The team must consolidate the current duplicates and reduce the chance that future inbound payloads create the same issue.

Which actions should the team take? (Choose two options.)

- A. Use Duplicate CI Remediator to select a surviving CI and merge required data and relationships
- B. Review and correct the applicable identification rules for future inbound data
- C. Change reconciliation priorities to ensure that each source creates a separate CI
- D. Delete duplicate CI records manually without reviewing their relationships

ANSWER: A B

Explanation:

Use Duplicate CI Remediator to select a surviving CI and merge required data and relationships is correct for resolving the current duplicate records. The team should preserve the CI with the most appropriate authoritative data while carrying forward valid relationships and attribute values needed for service context.

Review and correct the applicable identification rules for future inbound data is correct for prevention. Identification rules determine whether inbound source data matches an existing CI, so they must use stable, sufficiently unique criteria for the network-device class. Changing reconciliation rules alone does not correct an identification failure, and deleting records manually can discard relationships and valuable data.

QUESTION NO: 20

An Enterprise Architect of a financial services company is working across the enterprise and wants to track their capabilities. Which CSDM 5 domain is used? (Choose 1 option)

- A. Foundation
- B. Build and Integration (Build)
- C. Service Consumption (Sell/Consume)
- D. Design and Planning (Design)
- E. Service Delivery (Manage Technical)

ANSWER: D

Explanation:

Design and Planning (Design) is the CSDM 5 domain used to model and track enterprise capabilities. This domain supports the strategic and architectural perspective of the organization: understanding what the business does, how it creates value, and how its business architecture relates to applications, information, and services. A business capability is a stable, outcome-oriented description of an organization's ability, such as lending, fraud management, or customer onboarding. Because capabilities are independent of a particular organizational structure, process implementation, or technology product, they give an enterprise architect a consistent way to assess the enterprise and plan transformation.

In a financial-services organization, capability data is especially useful for connecting strategic objectives and regulatory priorities to the applications and services that enable them. It supports architecture roadmaps, investment and portfolio decisions, application rationalization, dependency analysis, and identification of the impacts of a change across the enterprise. CSDM places these planning-oriented architecture concepts in Design and Planning so they can be governed separately from the operational records used to manage technical service delivery. ServiceNow's CSDM documentation describes the model's domains and their intended use cases, including the architecture and planning concepts represented by this domain. See the [ServiceNow CSDM documentation](#) and the [ServiceNow CSDM overview](#).

QUESTION NO: 21

(Choose two)

A CMDB Administrator is leveraging CI data as part of an Integrated Risk Management (IRM) implementation and the Entity Scoping process. The Administrator wants to leverage the CSDM Data Foundations Dashboard playbooks under the Run tab.

Which CSDM relationships are leveraged using the CSDM playbooks?

- A. Business Applications that have their relationships to Information Objects
- B. Locations that have established parent records
- C. Business Applications that have relationships to Application Services

D. Logical CIs that have relationships with Information Objects

ANSWER: A D

Explanation:

Business Applications that have their relationships to Information Objects and **Logical CIs that have relationships with Information Objects** are the relationships used by the relevant Run playbooks for IRM entity scoping. Information Objects provide the CSDM representation of data that an organization manages, such as regulated, confidential, personal, or financial information. Relating a Business Application to an Information Object establishes the business context for that data: which application creates, uses, or manages it. This allows IRM teams to scope risk and compliance activities around the applications responsible for important information.

Relating Logical CIs to Information Objects adds the necessary technical context. It connects the information to the logical technology components that store, process, or otherwise support it, enabling a traceable path from data-oriented risk requirements to the underlying technical estate. Together, these relationships support data-centric entity scoping, risk assessment, control evaluation, and prioritization of remediation work. They also improve the completeness and usability of CMDB data for operational governance use cases rather than treating configuration data as isolated inventory. ServiceNow describes CSDM as the common framework for modeling and relating service, application, and technology data; see the [Common Service Data Model overview](#) and the [ServiceNow CSDM documentation](#).

QUESTION NO: 22

A service operations team needs a CMDB group for a fixed set of database servers that support a regulatory reporting process. Membership is determined by an approved list maintained by the service owner and must not change automatically when CI attribute values change.

Which CMDB group approach should the team use?

- A. Manually add the approved database server CIs to a named CMDB group
- B. Create a query-based CMDB group using the server class only
- C. Create reconciliation rules that assign matching CIs to the group
- D. Create a new CI class for the regulatory reporting servers

ANSWER: A

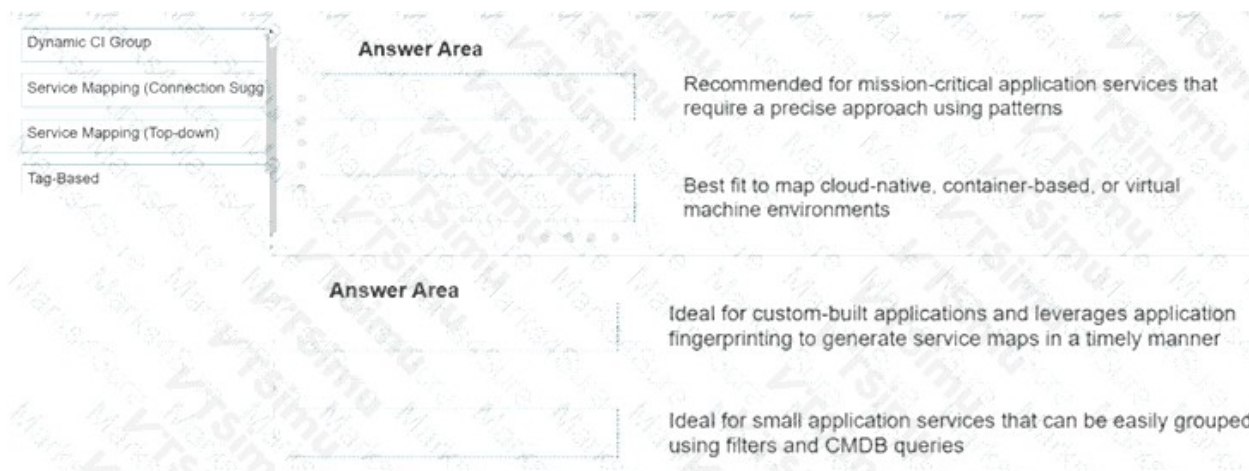
Explanation:

The team should **manually add the approved database server CIs to a named CMDB group**. This creates explicitly managed membership and is appropriate when the group represents a curated population maintained by a service owner.

A query-based CMDB group is suitable when membership should be calculated dynamically from CI attributes, such as class, environment, location, or assignment group. Reconciliation rules control source updates and do not define CMDB group membership.

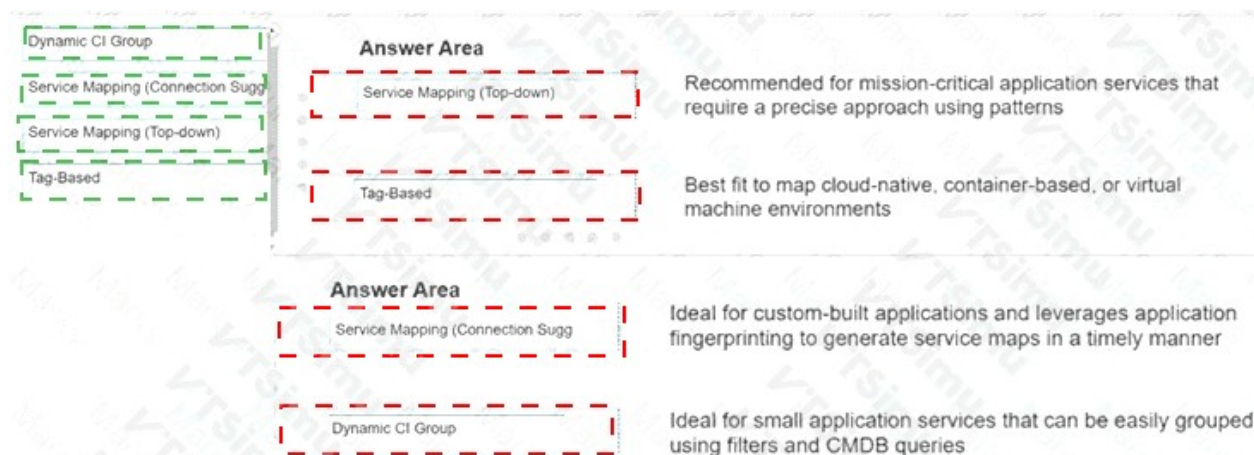
QUESTION NO: 23 - (DRAG DROP)

Drag and drop the application service type to the best description.



ANSWER:

Answer:



Answer:

Explanation:

Service Mapping (Top-down) → Recommended for mission-critical application services that require a precise approach using patterns
 Tag-Based → Best fit to map cloud-native, container-based, or virtual machine environments
 Service Mapping (Connection Suggestions) → Ideal for custom-built applications and leverages application fingerprinting to generate service maps in a timely manner
 Dynamic CI Group → Ideal for small application services that can be easily grouped using filters and CMDB queries
 In Data Foundations (CMDB + CSDM), an Application Service should be created using an approach that balances accuracy, speed, and maintainability, because the service map becomes the foundation for impact analysis, service visibility, and operational workflows. Service Mapping (Top-down) is best for mission-critical services because it uses structured entry points and patterns to precisely discover infrastructure, dependencies, and communication paths. This produces high-fidelity maps suitable for strict governance, controlled change, and reliable outage impact analysis. Tag-Based mapping is best for cloud-native and dynamic environments (containers, elastic virtual machines, and cloud resources) because tags/labels are the most stable identifiers in rapidly changing estates. It aligns well with modern cloud governance where ownership, environment, and service affiliation are expressed through standardized tagging strategies. Service Mapping (Connection Suggestions) fits custom-built applications because it leverages observed connection data and application fingerprinting to propose likely dependencies. This accelerates service map creation when traditional pattern-first modeling would take longer, helping teams reach usable service visibility faster while still allowing validation and refinement. Dynamic CI Group is ideal for smaller services where a full discovered map may be unnecessary. It groups CIs using filters and CMDB queries, providing a lightweight way to define “what’s in the service” for reporting, ownership, and basic impact without heavy mapping overhead.

Explanation:

Application service creation methods should be selected according to the characteristics of the environment and the level of service visibility required. **Service Mapping (Top-down)** is the appropriate method for a mission-critical application service that needs a precise, pattern-driven map. Starting from defined entry points, Service Mapping discovers the components and dependencies that support the service. This structured approach produces the detailed dependency information needed for

reliable operational visibility, impact analysis, and service-aware change decisions. ServiceNow describes Service Mapping as a way to discover and map application-service dependencies; see the [ServiceNow Service Mapping product page](#).

Tag-Based mapping is suited to cloud-native, container-based, and virtual-machine environments because standardized tags can identify related cloud resources even as those resources are created, scaled, moved, or retired. Tags provide a practical service-association mechanism in dynamic infrastructure and support consistent ownership and environment identification.

Service Mapping (Connection Suggestions) is appropriate for custom-built applications where observed connection information and application fingerprinting can accelerate initial map creation. It provides a timely way to assemble a service map from the application's actual communications, giving teams a useful foundation that can then be reviewed and refined.

Dynamic CI Group is the lightweight choice for a small application service whose member CIs can be defined reliably through CMDB filters and queries. Rather than requiring a full dependency map, the group dynamically represents the relevant configuration items based on maintained criteria. This supports a manageable service definition while retaining the CMDB-based context essential to CSDM. More CMDB context is available on the [ServiceNow CMDB product page](#).

QUESTION NO: 24

A CMDB Administrator, viewing the CMDB Data Foundations Dashboard, notices the Unique Locations Result percentage is low.

What is the recommended process from the associated playbook to correct this issue?

- A. Use the Duplicate CI Remediator to merge the duplicate location records
- B. Retain the location that matches the organization 's standard naming convention, and delete the duplicate without further validation
- C. Review both locations, update CIs with the correct location, and delete the duplicate location
- D. Keep both locations as either can be used as a valid alternate location

ANSWER: C

Explanation:

Review both locations, update CIs with the correct location, and delete the duplicate location is the recommended process because the Unique Locations metric identifies cases in which multiple location records represent the same real-world location. Resolving the issue requires first validating the records to establish which location is authoritative. The administrator should then identify the configuration items that reference the duplicate record and update those CI records to reference the validated, correct location. This preserves accurate CI-to-location relationships and ensures that reporting, ownership, support processes, and location-based analysis continue to use a consistent location value. Only after those references have been remediated should the redundant location record be removed. This sequence is important because location values are reference data in the CMDB: correcting the duplicate record alone does not automatically ensure that all affected CIs point to the intended authoritative location. The Data Foundations Dashboard playbooks are designed to guide this kind of data-quality remediation by validating the source data, correcting dependent CI relationships, and then eliminating the duplicate data record. See ServiceNow's [CMDB Data Foundations Dashboard documentation](#) and [CMDB Data Foundations documentation](#).

QUESTION NO: 25

The Configuration Management team wants to confirm that all servers in the CMDB actually exist in the data center. Which CMDB Data Manager policy type would the team create? (Choose 1 option)

- A. Certification
- B. Delete
- C. Archive
- D. Retire

ANSWER: E

Explanation:

Attestation is the appropriate CMDB Data Manager policy type because the required outcome is a periodic, accountable confirmation that a server CI remains valid and physically exists in the environment. An attestation policy sends review tasks to designated CI owners, managers, or other responsible stakeholders. Those reviewers assess the CI and provide an explicit response, creating an auditable record of the validation activity. This human review is particularly useful for confirming real-world conditions that may not be fully established by automated discovery alone, such as whether equipment remains deployed in a particular data center.

For this use case, the team can target the relevant server CI classes and assign the attestation to people with responsibility for the infrastructure. Their responses help maintain CMDB trust by identifying records that are still present and valid, as well as records needing follow-up or lifecycle remediation. The process supports governance by establishing ownership, review cadence, and evidence that the organization has validated its configuration data. ServiceNow describes CMDB Data Manager policies as tools for managing CMDB data quality and lifecycle activities, including attestation-based reviews. See the [ServiceNow CMDB Data Manager documentation](#) and the [ServiceNow attestation policy guidance](#).