ServiceNow.CIS-DF-JPN.v2026-08-14.q48

試験コード：	CIS-DF-JPN
試験名称：	Certified Implementation Specialist - Data Foundations (CMDB and CSDM) (CIS-DF日本語版)
認証ベンダー：	ServiceNow
無料問題の数：	48
バージョン：	v2026-08-14
ページの閲覧量：	102
問題集の閲覧量：	481

https://www.jpnshiken.com/shiken/ServiceNow.CIS-DF-JPN.v2026-08-14.q48.html

質問: **1**
ハードウェアCIクラスの次の識別ルールが定義されています

2 つの新しい CI レコードが CMDB のハードウェア クラスにインポートされます。
CI1: この CI レコードの名前は、CMDB 内の既存の CI レコードの名前と一致します。
CI2: この CI レコードの IP アドレスは、CMDB 内の既存の CI レコードの IP アドレスと一致します。
識別ルールとインポートされた CI レコードに基づいて正しいのはどれですか?
A. CI1 は新しいレコードとして挿入され、CI2 は一致するレコードで更新されます。
B. CI1とCI2は両方とも新しいレコードとして挿入されます
C. CI1は一致するレコードで更新され、CI2は新しいレコードとして挿入されます。
D. CI1とCI2は両方とも一致するレコードで更新されます
正解: **(**正解を表示します**)**
This question tests understanding of how the Identification and Reconciliation Engine (IRE) evaluates incoming CI data against Identification Rules and their priority order in ServiceNow.
From the identification rule shown:
Serial number (+ type) # Priority 100
Serial number # Priority 200
Name (Hardware) # Priority 300
MAC address + name (Network Adapter) # Priority 400
For a CI to be identified and matched, the incoming record must satisfy one complete identifier entry exactly as defined for that class.
CI1 (Name match only)
Although the name matches an existing Hardware CI, name alone is a low-priority identifier (300) and is not sufficient to uniquely identify a Hardware CI unless no higher-priority identifiers exist and the identifier entry criteria are fully satisfied. In practice, Hardware identification relies on serial number-based identifiers, not name-only matching, to avoid false positives. Therefore, CI1 cannot be confidently matched and is inserted as a new record.
CI2 (IP address match)

IP address is not part of any Hardware identification rule shown. IP address is typically used for discovery correlation or network relationships, not as a primary Hardware identifier. Since no identifier entry includes IP address, CI2 does not match any valid identification rule and is also inserted as a new record.
Because neither CI satisfies a valid identifier entry, both records are inserted as new CIs.

質問: **2**
(2つ選択してください)
Windows Server (cmdb_ci_win_server) クラスに対して、ServiceNow、Altiris、および SCCM 用に以下の調整ルールが設定されました。

Discovery Source	Class	Priority
ServiceNow	Windows Server [cmdb_ci_win_server]	100
Altiris	Windows Server [cmdb_ci_win_server]	200
SCCM	Windows Server [cmdb_ci_win_server]	300

どの記述が正しいですか？
A. Altirisの検出ソースで収集されたデータは、SCCMによってWindows Serverテーブルに挿入されたレコードを更新できます。
B. ServiceNowの検出ソースで収集されたデータは、Windows Serverテーブルに新しいレコードを挿入できますが、AltirisまたはSCCMによって作成されたレコードを更新することはできません。
C. SCCM の検出ソースで収集されたデータは、Windows Server テーブルに新しいレコードとして挿入できます。
D. SCCM の検出ソースで収集されたデータは、最も高い優先順位番号を持つため、Windows Server テーブル内の任意のレコードを更新できます。
正解: **A,C (**コメントを発表する**)**
This question tests understanding of reconciliation source priority in the Identification and Reconciliation Engine (IRE) in ServiceNow.
In reconciliation rules, lower numeric values represent higher priority. Therefore, the priority order is:
ServiceNow (100) - highest authority
Altiris (200)
SCCM (300) - lowest authority
Why A is correct
Because Altiris (200) has higher priority than SCCM (300), data from Altiris can update records originally inserted by SCCM. This is exactly how reconciliation precedence works-higher-priority sources can overwrite lower-priority ones.
Why C is correct
SCCM, even though it has the lowest priority, is still an authorized discovery source. It can insert new records into the Windows Server table when no existing CI is identified. Priority only affects updates, not the ability to create records.
Why B is incorrect
ServiceNow (priority 100) can update records from Altiris and SCCM because it has the highest priority. The statement incorrectly claims it cannot.
Why D is incorrect
SCCM does not have the highest authority. A higher numeric value means lower priority, so SCCM cannot update records created by higher-priority sources.

質問: **3**
構成管理チームは、CMDB内のすべてのサーバーがデータセンター内に実際に存在することを確認したいと考えています。チームはどのタイプのCMDBデータマネージャポリシーを作成しますか？(1つ選択してください)
A. 認定
B. 削除

C. アーカイブ
D. 削除
E. 証明
正解: **(**正解を表示します**)**
Comprehensive and Detailed Explanation (200-300 words)
Within ServiceNow Data Foundations, CMDB Data Manager provides multiple policy types to support governance, data quality, and lifecycle management of configuration items (CIs). The scenario described- confirming that servers recorded in the CMDB physically exist in the data center-is a classic example of existence validation and ownership confirmation, which is exactly the purpose of an Attestation policy.
An Attestation policy is designed to request a human validation from a responsible individual or group (such as a data center manager, platform owner, or infrastructure team). The policy generates attestation tasks that require reviewers to explicitly confirm whether a CI is valid, accurate, and still exists. This aligns directly with CMDB governance best practices and ITIL 4 Service Configuration Management, where periodic verification ensures trust in the CMDB as a system of record.
The other policy types do not meet this requirement:
* Certification is typically used to validate compliance with defined data standards (e.g., mandatory fields populated), not physical existence.
* Delete, Archive, and Retire are lifecycle actions, used after a CI has already been identified as obsolete or no longer required.
* None of these options involve human confirmation of real-world existence.
From a CSDM and Data Foundations perspective, attestation supports:
* CMDB accuracy and credibility
* Audit and regulatory compliance (especially critical in financial services)
* Clear accountability for CI ownership and validation
Therefore, when the goal is to confirm that servers actually exist, the correct and fully aligned CMDB Data Manager policy type is Attestation (E).

質問: **4**
構成マネージャーは CMDB の実装を計画しています。
規定の CSDM 展開順序はどれですか?
A. 初期、開発中、定義済み、管理済み
B. アーキテクチャ、ビジネス、技術、ガバナンス
C. 開始、計画、実行、提供、終了
D. 這う、歩く、走る、飛ぶ
正解: **(**正解を表示します**)**
The Common Service Data Model (CSDM) prescribes an incremental, maturity-based rollout approach to reduce risk and ensure sustainable adoption. The recommended order is Crawl, Walk, Run, Fly, which aligns implementation effort with increasing organizational capability and value realization.
Crawl focuses on foundational data hygiene: core CI classes, identification rules, reconciliation, basic Discovery ingestion, and CMDB Health basics.
Walk introduces service context, including Business Services, Application Services, and relationships that enable impact analysis for Incident and Change.
Run expands into operational excellence with Service Mapping, service offerings, advanced governance, and process automation.
Fly represents optimization and scale, leveraging analytics, AI/ML, proactive operations, and cross-domain integration (e.g., SecOps, APM, CSM).
This progression ensures teams do not over-model early or introduce complexity before data quality and governance are established. The other options describe generic project lifecycles or organizational categorizations, not the CSDM-recommended adoption path.
Therefore, the correct answer is D - Crawl, Walk, Run, Fly.

質問: **5**

CMDB データ マネージャ ポリシーを設定し、割り当てタイプを 「ユーザー グループ フィールド」に設定してタスク割り当てを指定するときに使用できるデフォルトのユーザー グループはどれですか? (2 つのオプションを選択)
A. グループ所有
B. サポートグループ
C. グループによって管理
D. 割り当てグループ
正解: (正解を表示します)
In ServiceNow Data Foundations, specifically within CMDB Data Manager, policies are used to enforce data quality, completeness, and governance across configuration items (CIs). When defining a Data Manager policy, administrators can configure task creation for non-compliant records and specify how those tasks are assigned. One key configuration is the Assignment type, where "User Group Field" allows assignment to a group dynamically based on a group reference field on the CI record.
Out of the box, ServiceNow provides two default and universally available group reference fields for this purpose:
* Assignment Group
* Managed By Group
These two fields are considered core CMDB governance fields and are consistently available across most CI classes. They align directly with ITIL 4 Service Configuration Management and CSDM accountability models, ensuring that remediation tasks are routed to teams responsible for operational ownership or technical management of the CI.
* Assignment Group (Correct - D) This is the most commonly used operational field in ServiceNow. It represents the team responsible for working on tasks related to the CI and is fully supported by CMDB Data Manager for automated task assignment.
* Managed By Group (Correct - C) This field represents the group accountable for managing the CI throughout its lifecycle. It is a standard CMDB field and is explicitly supported by Data Manager policies for governance-driven task routing.
The other options are not default Data Manager assignment fields:
* Owned by Group is often added through extensions or governance customization and is not guaranteed to exist across all CI classes.
* Support Group is a CSDM and ITSM operational concept, commonly derived or mapped, but it is not a default group reference field exposed for CMDB Data Manager task assignment.
From a Data Foundations and CSDM governance perspective, using Assignment Group and Managed By Group ensures consistent accountability, scalable automation, and alignment with CMDB best practices.

質問: **6**
(2つ選択)
CMDB管理者は、統合リスク管理(IRM)の実装およびエンティティスコープ設定プロセスの一環として、CIデータを活用しています。管理者は、[実行]タブにあるCSDMデータ基盤ダッシュボードのプレイブックを活用したいと考えています。
CSDMプレイブックを活用することで、どのCSDM関係性が利用されるのでしょうか?
A. 情報オブジェクトと関係を持つビジネスアプリケーション
B. 親の記録が確立されている場所
C. アプリケーションサービスと関係のあるビジネスアプリケーション
D. 情報オブジェクトと関係を持つ論理的なCI
正解: (正解を表示します)
The Run tab in the CSDM Data Foundations Dashboard focuses on enabling operationalized, risk-aware use cases, including Integrated Risk Management (IRM) and Entity Scoping. These use cases require organizations to understand what data is processed, where it resides, and which technical components are involved, rather than only service impact for ITSM.
Information Objects play a central role at this stage.
Option A is correct because Business Applications related to Information Objects allow organizations to identify what types of data (PII, PCI, PHI, regulated data) are processed by each business application. This relationship is essential for risk classification, regulatory compliance, and audit scoping in IRM. Without it, risk assessments lack data sensitivity context.

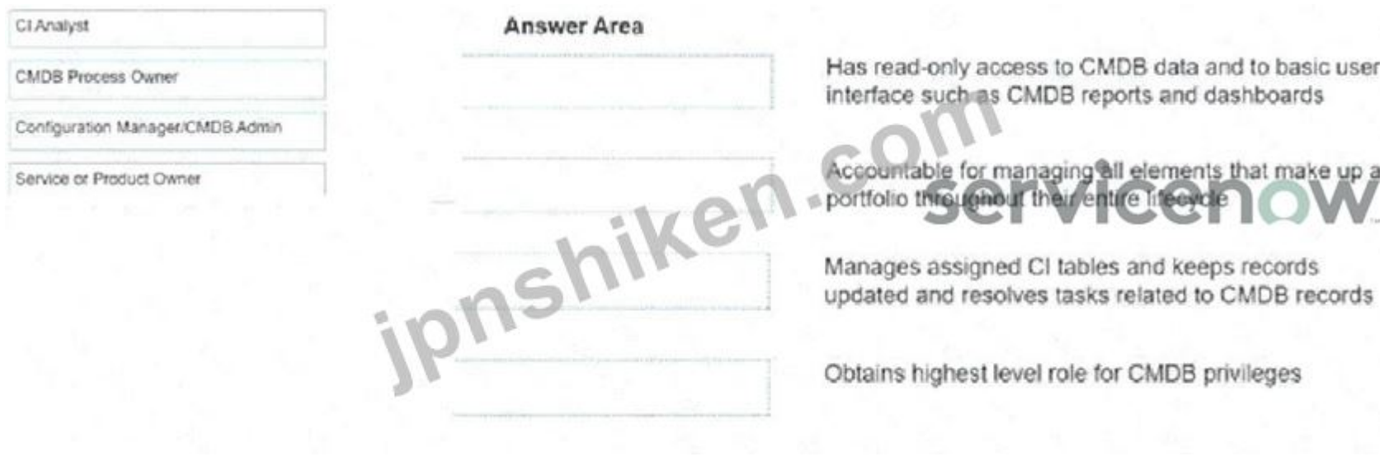
Option D is also correct because Logical CIs (such as databases, schemas, or data stores) related to Information Objects establish where sensitive data is stored or processed at a technical level. This enables IRM to trace risk from business context down to technical exposure, supporting control testing, issue management, and remediation prioritization.
Option B (Location hierarchy) supports foundational data quality but does not directly enable risk or entity scoping. Option C (Business Applications to Application Services) is critical for service impact and Change
/Incident Management, but it is more aligned to service operations rather than risk and data-centric scoping, which is the focus of the Run playbooks for IRM.
Therefore, the correct answers are A and D, as they directly support IRM entity scoping, regulatory analysis, and risk visibility through CSDM-aligned data modeling.

質問**: 7**
CMDB 管理者がサービス グラフ コネクタをインストールし、スクリプト変換をカスタマイズしました。
スクリプト変換のデフォルト定義が更新された場合、後続のアップグレードでは何が起こりますか?
A. アップグレードが停止し、エラーが報告されます
B. スキップされた変更が作成され、スクリプト変換定義には変更が加えられません。
C. サービスグラフコネクタのアップグレードが開始できません
正解**: B (**コメントを発表する**)**
In ServiceNow, Service Graph Connectors deliver data ingestion patterns using protected, upgrade-safe artifacts, including script transforms. When a customer customizes a script transform provided by a Service Graph Connector, ServiceNow follows standard update set and upgrade behavior to protect customer customizations.
During a subsequent upgrade, if the out-of-box (default) script transform definition changes, ServiceNow does not overwrite the customized version. Instead, the platform records a skipped change, indicating that an update was available but intentionally not applied due to a local customization. This ensures customer- specific logic is preserved while still maintaining transparency about what changed in the newer release.
Option A is incorrect because upgrades do not halt due to customized transforms. Option C is also incorrect because Service Graph Connector upgrades proceed normally; they do not refuse to start because of customizations.
This behavior aligns with Data Foundations best practices: avoid modifying OOTB content when possible, but when customization is necessary, ensure it is protected during upgrades. Administrators should review skipped changes after upgrades to decide whether to manually adopt new OOTB logic.
Therefore, the correct answer is B - A skipped change is created and no change is made to the script transform definition.

質問**: 8**
組織は、ハードウェアCIの保証有効期限など、検出不可能な属性を管理する必要があります。これらの属性は、自動検出ツールでは更新されません。
すべての CI に対してこれらの属性が正確に維持されることを保証する方法は何ですか?
A. CMDB調整エンジンを使用して属性を更新します
B. 検出できない属性専用の新しいCIクラスを作成します。
C. スケジュールされたデータインポートを使用して外部ソースから属性を更新します
正解**: (**正解を表示します**)**
In Data Foundations, "Ingest" is not limited to Discovery. It includes establishing reliable sources and mechanisms to populate and maintain CMDB attributes-especially those that Discovery cannot provide.
Warranty expiration dates are a classic example of non-discoverable data: they typically come from procurement systems, asset repositories, vendors, or contract tools rather than from device interrogation.
The correct approach is to maintain these attributes via a scheduled data import from an authoritative external source. This ensures ongoing accuracy and consistency at scale. A scheduled import allows you to define a repeatable integration pattern, apply data quality checks, and keep warranty data current as vendors and procurement records change-without relying on manual updates that inevitably drift over time.

Option A is incorrect because the CMDB Reconciliation Engine resolves conflicts between multiple data sources and applies precedence rules; it does not create the warranty values out of thin air. Reconciliation helps decide which source wins when updates occur, but you still need a source feed providing the warranty attribute values. Option B is also not recommended because creating a new CI class just to hold non- discoverable attributes fragments the model and complicates reporting and processes. Best practice is to keep the attributes on the appropriate hardware CI classes (or related normalized structures, when applicable) and manage them through governed ingestion from authoritative systems. Therefore, C is the best method.

質問: **9**
ServiceNowの新規顧客が、自社のCMDBをサポートするために構成管理チームを編成している。
各役割を対応する職務内容にドラッグしてください。



正解:

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

質問: **10**
構成マネージャーは、CMDB 内の同じクラスおよび同じクラス属性セットを更新する権限を持つ複数のデータ ソースを構成しました。
特定のクラスまたはクラス属性セットの信頼できるソースとなるデータ ソースを制御するために、構成マネージャーで何ができますか?
A. 照合ルール内の各データ ソースに優先順位を割り当てる
B. データ ソースの更新を正しい順序で手動で実行します
C. 特定の期間でデータ更新ルールを設定する
D. 識別ルール内の各データ ソースに実行順序を割り当てます。
正解: **(**正解を表示します**)**
In ServiceNow, controlling source precedence when multiple authorized data sources update the same CI attributes is a core responsibility of Identification and Reconciliation Engine (IRE) governance.
The correct and supported method is to assign priority to each data source in reconciliation rules.
Reconciliation rules determine which source wins when multiple sources attempt to update the same attribute on a CI. By defining source precedence, the Configuration Manager ensures that the most authoritative system of record (for example, Discovery over manual imports, or HR systems over spreadsheets) consistently controls specific attributes or classes.
Option B is incorrect because manually sequencing data source runs is unreliable, does not scale, and violates Data Foundations best practices. Option C only controls how often data is refreshed, not which source is authoritative. Option D is incorrect because identification rules are used to uniquely identify CIs-not to control attribute-level precedence.
Using reconciliation rules provides deterministic, auditable, and automated control, which is essential for maintaining CMDB trust and avoiding data conflicts.
Therefore, the correct answer is A - Assign a priority to each data source in reconciliation rules.

質問: **11**
インポート セットと変換マップを使用してデータを CMDB に統合する場合、データが識別および調整エンジン (IRE) を介して処理されるようにするために追加されるスクリプトの種類はどれですか。
A. オンビフォア
B. 完了時
C. オンアフター
D. 開始時
正解: **C (**コメントを発表する**)**
When using Import Sets and Transform Maps to ingest data into the CMDB, it is critical that records are processed through the Identification and Reconciliation Engine (IRE) to prevent duplicates and enforce source precedence. In ServiceNow, this is achieved by invoking the IRE after the transform logic has completed.
The onAfter transform script is the correct place to call the IRE API. At this stage, the transformed data has already been mapped and prepared, allowing the IRE to correctly identify whether a CI already exists and reconcile updates according to defined rules.
The onBefore and onStart scripts execute too early-before data mapping is complete-making them unsuitable for IRE processing. The onComplete script runs after the entire import job finishes and is not intended for per-record CI identification and reconciliation.
Because Import Sets can bypass IRE if not configured correctly, using an onAfter script is a critical Data Foundations safeguard when this ingestion method is chosen.
Therefore, the correct answer is C - onAfter.

質問: **12**
ユーザーのエンドポイントデバイスはCMDBにインポートされ、コンピュータCI [cmdb_ci_computer] の 割り当て先 [assigned_to]」フィールドに入力されます。資産チームは、構成アナリストに対し、関連する資産の 割り当て先」フィールドに入力するよう依頼します。
これを自動的に実現するために、構成アナリストはどのようなアクションを実行しますか?
A. 資産レコードの 割り当て先」フィールドを非表示にし、関連するCIにドットウォークして 割り当て先」の値を取得する新しいフィールドを作成します。

B. コンピュータクラスの 割り当て先」フィールドへの挿入または更新に基づいて、資産の 割り当て先」フィールドにデータを入力するスクリプトを使用するように、コンピュータテーブルにビジネスルールを設定します。
C. アセット CI フィールド マッピング モジュールを使用して、アセットと関連付けられた CI 間で 割り当て先」の値を複製する新しいルールを作成します。
正解: **C (**コメントを発表する**)**
Data Foundations strongly encourages using out-of-box synchronization mechanisms rather than custom scripting whenever possible, to reduce technical debt and improve upgrade safety. In this scenario, endpoint data is being ingested into the CI record, and the Asset team needs the corresponding Asset record kept in sync automatically for their asset management processes.
The recommended approach is to use Asset CI Field Mapping, which is designed specifically to map and synchronize fields between an Asset and its related CI. Creating or updating a mapping rule for "Assigned to" allows the platform to replicate the value in a controlled, supportable way without writing scripts. This approach aligns with best practices: standard configuration, clear governance, and minimal custom code.
Option B (a scripted business rule) can work technically, but it introduces maintenance risk, testing overhead, and potential upgrade fragility-exactly the kind of technical debt Data Foundations warns against when an OOTB capability exists. Option A (dot-walking) changes the user experience and does not truly populate the Asset field; it only displays a derived value and may not meet Asset team needs for reporting, exports, integrations, or lifecycle workflows that expect the Asset field to be set. Therefore, the best automated and best-practice solution is Asset CI Field Mapping.

質問: **13**
(2つ選択してください)
CMDB管理者は、ServiceNowでモノのインターネット(IoT)センサー用の新しいCIクラスを作成する必要があります。
この活動を行う上で推奨される手順は何ですか？
A. CMDB CIクラスモデルストアアプリケーションをインストールまたは更新し、クラスが既に存在しないことを確認してください。
B. 使用されていないクラスを削除し、新しいクラスに置き換えます
C. 既存のクラスを変更する
D. 適切な親クラスの下に新しいクラスを追加します
正解: **A,D (**コメントを発表する**)**

質問: **14**
構成マネージャーは、ポリシー タイプを活用して、CI の存在を検証するためのタスクの作成と割り当てを自動化する必要があります。
この目標を達成するにはどのポリシー タイプを使用する必要がありますか?
A. 認定
B. 削除
C. 引退
D. 証明書
正解: **(**正解を表示します**)**
In ServiceNow, validating whether Configuration Items (CIs) still exist is a core CMDB governance activity.
Over time, environments change rapidly-servers are decommissioned, cloud resources are torn down, and applications are replaced. If existence validation is not enforced, the CMDB quickly fills with obsolete or
"ghost" CIs.
Attestation policies are specifically designed to address this need. An attestation policy automatically generates and assigns tasks to responsible users or groups, asking them to confirm that a CI still exists and is still relevant. This process focuses on acknowledgment rather than deep data validation, making it lightweight and scalable across large CMDBs.
Certification policies (Option A) are used when specific attributes must be validated, such as lifecycle status, support group, or environment. While important for data correctness, certification is not intended solely to confirm CI existence. Delete (Option B) and Retire (Option C) policies are lifecycle actions that remove or transition records, but they do not validate existence before taking action.

Attestation integrates cleanly with CMDB Workspace, assigns tasks automatically, and supports auditability- ensuring accountability for CI ownership. This makes it the correct and Data Foundations-aligned policy type for validating CI existence.
Therefore, Option D - Attestation is the correct answer.

質問: **15**
CMDB 管理者は、インシデント フォームからアクセス可能な CI 参照フィールドなど、CI 参照フィールドにプリンシパル クラスの CI のみが表示されるようにしたいと考えています。
CMDB 管理者はプリンシパル クラスをどこで指定しますか?
A. CMDBワークスペース
B. CI クラス マネージャー
C. CMDBデータマネージャー
D. システムプロパティ
正解: **(**正解を表示します**)**
Within Data Foundations, "Configuration" includes designing the CMDB data model in a way that supports operational usability-especially for ITSM processes like Incident and Change. One common usability challenge is that CI reference fields (such as the Incident "Configuration item" field) can become cluttered and confusing if they allow selection from every CI class, including highly technical or rarely used classes. To address this, ServiceNow uses the concept of Principal Classes: a curated subset of CI classes that are considered primary for operational selection and reporting.
Principal Classes are designated in the CI Class Manager, because this is where administrators configure and manage CI class behaviors and class-level settings. Defining Principal Classes at the class-management layer ensures that the platform can consistently apply this constraint across CI reference fields and related experiences. This aligns with Data Foundations goals: improving data quality and process outcomes by making it easier for users to select the correct CI, which in turn increases accuracy for assignment, routing, impact analysis, and analytics.
The other options do not fit the responsibility. CMDB Workspace is primarily used for exploring and analyzing CMDB data, not configuring class-level constraints. CMDB Data Manager focuses on governance automation (like lifecycle policies and bulk remediation operations), not CI class designation rules. System Properties may influence global behaviors, but Principal Class designation is a modeling configuration handled through CI class configuration, which is why CI Class Manager is the correct location.

質問: **16**
CMDB管理者は、ServiceNow本番インスタンス内に2つの動的照合ルールを設定しました。「Server」クラスには、RAMフィールドの値が最大値となる動的照合ルールが設定されています。「Windows Server」クラスには、RAMフィールドの報告数が最多となる動的照合ルールが設定されています。

Discovery Source	RAM (MB)
Tivoli	4,096
ServiceNow	4,096
LANDesk	2,048
Altiris	8,192

マルチソース CMDB の上記のデータを考慮すると、サーバー CI の RAM の CMDB にはどの値が追加されますか?
A. 2,048 MB
B. 8,192 MB
C. 4,096 MB
正解: **(**正解を表示します**)**
This question hinges on understanding class-specific Dynamic Reconciliation Rules in CMDB 360 / Multisource CMDB within ServiceNow.
Although two different dynamic rules are configured, the rule applied depends on the CI class being evaluated:
For the Server class, the configured rule is Largest value for the RAM attribute.
For the Windows Server class, the rule Most reported would apply-but only if the CI were evaluated as a Windows Server.
The question explicitly asks for the resulting RAM value for a Server CI, not a Windows Server CI.

Therefore, the Largest value rule governs the outcome.
Looking at the multisource values:
2,048 MB (LANDesk)
4,096 MB (Tivoli)
4,096 MB (ServiceNow)
8,192 MB (Altiris)
Under the Largest value dynamic reconciliation rule, the IRE selects the maximum numeric value, regardless of how frequently it is reported or which source provided it.
The Most reported logic (which would result in 4,096 MB) does not apply here because that rule is configured for a different class (Windows Server).
This scenario illustrates an important CMDB 360 principle: Dynamic reconciliation is evaluated per class, and child-class rules do not override parent-class rules unless the CI is actually classified under that child class.

質問: 17
構成管理ガバナンス チームは、従来の CMDB ステータス フィールドの利用から CSDM ライフサイクル ステータス フィールドへの移行を行っています。
どのテーブルを変更できますか?
A. ライフサイクルステージ
B. ライフサイクルマッピング
C. ライフサイクルステージのステータス
D. ライフサイクルコントロール
正解: (正解を表示します**)**
When organizations transition from legacy CMDB status fields (such as custom install status or operational status values) to CSDM-aligned lifecycle status fields, the goal is to map old values to standardized lifecycle stages without disrupting existing processes. In ServiceNow, this is achieved through the Life Cycle Mapping table.
The Life Cycle Mapping table is specifically designed to translate legacy or custom status values into CSDM lifecycle stages and statuses. This allows organizations to preserve historical data and integrations while progressively adopting CSDM standards. By modifying this table, administrators can define how existing status values correspond to CSDM lifecycle stages such as Plan, Build, Deploy, Operate, and Retire .
The Life Cycle Stages table (Option A) defines the standard stages themselves and should not be modified, as these are core to CSDM governance. Life Cycle Stage Status (Option C) defines valid statuses within a stage and is also part of the standardized model. Life Cycle Controls (Option D) enforce governance rules but do not perform value translation.
Therefore, to safely transition from legacy status fields to CSDM lifecycle statuses, the correct and supported approach is to modify the Life Cycle Mapping table, making Option B the correct answer.

質問: 18
開発チームは、アプリケーションを複数のサーバーにデプロイするプロジェクトに取り組んでいます。アプリケーションは顧客の個人情報(PIIおよびPCI)を保持するため、法令遵守のためにいくつかのセキュリティ要件を満たす必要があります。
開発チームは、監査を満たすために使用する情報を CSDM のどこに保存する必要がありますか?
A. テクノロジー管理サービス オファリング (テクニカル サービス オファリング) とグループ
B. ビジネスアプリケーションと情報オブジェクト

C. 顧客サービスの提供とデータベース
正解: **B (**コメントを発表する**)**
Within the Common Service Data Model (CSDM), regulatory, security, and compliance-related information- especially for PII and PCI-must be modeled at the business and information level, not at the infrastructure or service offering level. The correct location for this data is Business Applications combined with Information Objects.
Business Applications represent the logical applications that support business capabilities and processes.
Since compliance obligations (such as GDPR, PCI-DSS, or HIPAA) are assessed based on how the business uses data-not how many servers host the application-this is the correct anchor point for audit-relevant context.
Information Objects are explicitly designed to capture what data is processed, stored, or transmitted by an application, including data classifications such as PII, PCI, PHI, or confidential business data.
They allow organizations to document regulatory scope, retention rules, encryption requirements, and audit controls without overloading CI records or polluting infrastructure classes.
Option A is incorrect because Technical Service Offerings and Groups focus on operational support and service delivery, not regulatory data context. Option C is also incorrect because Customer Service Offerings describe how services are consumed, while databases are technical components; neither is the authoritative place for compliance definitions.
Therefore, Business Applications and Information Objects are the correct CSDM constructs to support audits and regulatory compliance, making Option B the correct answer.

質問: **19**
アプリケーションとサーバーの関係は何ですか?
A. アプリケーション > 使用者::使用 > サーバー
B. アプリケーション > 使用::使用元 > サーバー
C. アプリケーション > 実行::実行先 > サーバー
D. アプリケーション > 実行場所::実行場所 > サーバー
正解: **(**正解を表示します**)**
In Data Foundations (CMDB and CSDM), relationship modeling must reflect real operational dependency so that incident triage, change impact analysis, and service visibility remain accurate. When an Application is hosted on a Server, the standard hosting-style relationship used in CMDB relationship governance is expressed as "Runs on::Runs". This pairing represents the two directional descriptors of the same relationship type: from the application perspective it runs on the server, and from the server perspective it runs the application.
This matters because CMDB relationships are used by downstream capabilities (for example, dependency views, impact calculations, and governance rules). Using the correct out-of-box relationship descriptor pair ensures consistent reporting and prevents confusion when teams traverse relationships "upstream" and
"downstream." In addition, relationship governance rules and inheritance are commonly built around standard relationship types; using the correct "Runs on::Runs" semantics supports validation across subclasses (for example, specific application and server subclasses) without requiring custom relationship definitions.
The other options are either reversed ("Runs::Runs On"), represent different semantics ("Uses::Used by"), or do not align with the typical hosting relationship naming used for application-to-server hosting dependencies.
Therefore, the correct relationship expression between an Application and a Server is Application > Runs on::
Runs > Server.

質問: **20**
(2つ選択してください)
構成管理では、CMDB内のすべての構成アイテム(CI)のデータ品質を確保する必要があります。
CMDBヘルスダッシュボードには、構成アイテム(CI)のデータ品質に関するどのような領域が含まれていますか？
A. 格下げされたCI
B. アップグレードされたCI
C. CIが欠落しています
D. 古くなったCI

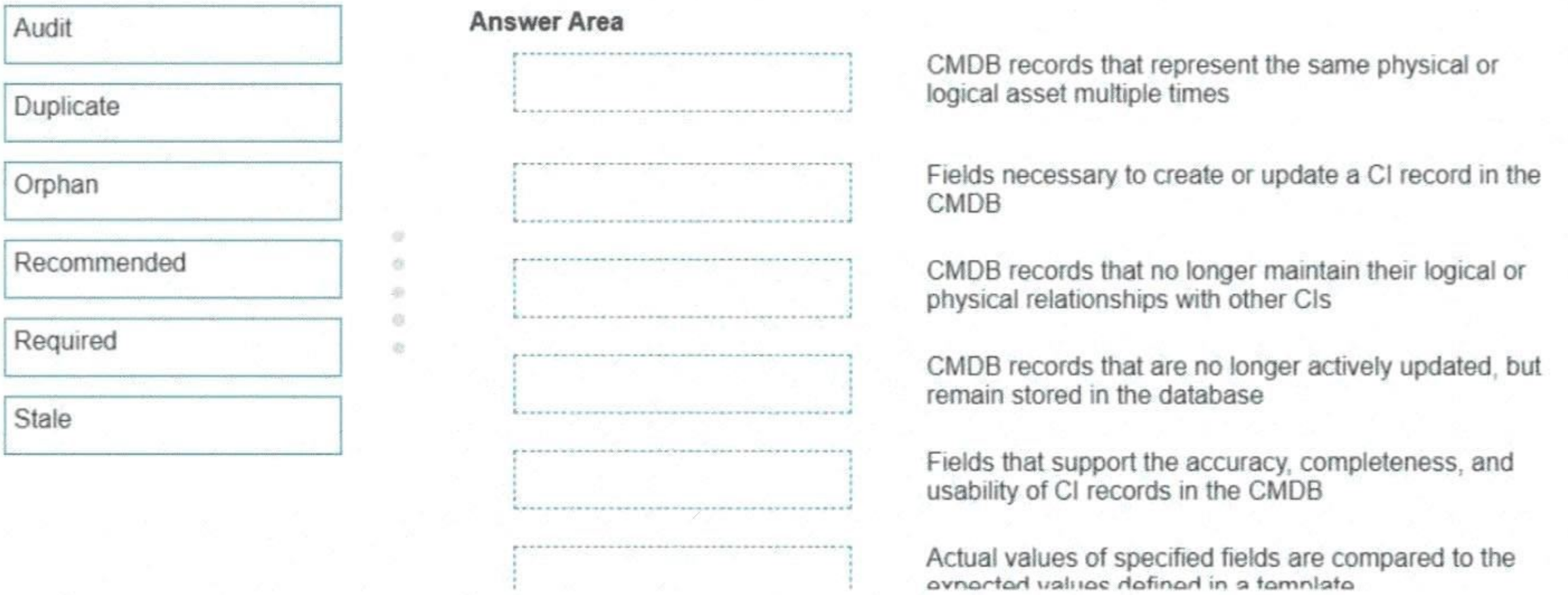
E. 重複するCI
正解: **D,E (**コメントを発表する**)**
The CMDB Health Dashboard is a central component of CMDB Data Foundations insight and governance. It measures and tracks data quality using well-defined health indicators that focus on the accuracy, relevance, and usability of CI data.
Two key data quality areas included in the dashboard are Stale CIs and Duplicate CIs.
Stale CIs (Option D) refer to configuration items that have not been updated within a defined time window.
These records are risky because they may no longer reflect the current state of the environment, leading to inaccurate impact analysis, poor change decisions, and misrouted incidents. Monitoring staleness helps organizations identify where discovery, integrations, or ownership processes are failing.
Duplicate CIs (Option E) occur when the same real-world asset or service is represented by multiple records.
Duplicates undermine trust in the CMDB, distort reporting, and break service mappings. The CMDB Health Dashboard highlights duplicate trends and integrates with de-duplication and remediation workflows to address them.
Options A (Downgraded CIs), B (Upgraded CIs), and C (Missing CIs) are not standard CMDB Health Dashboard quality dimensions. While "missing" data may be inferred through completeness checks, Missing CIs as a category is not directly tracked.
Therefore, the correct answers are D - Stale CIs and E - Duplicate CIs, which are core CMDB Health indicators used to maintain high-quality configuration data.

質問: **21**
CMDB 管理者は、CMDB 内の古い CI を最小限に抑えたいと考えています。
この情報はどの CMDB ヘルス ダッシュボード スコアカードに表示されますか?
A. 正確性
B. 完全性
C. コンプライアンス
正解: **C (**コメントを発表する**)**
In Data Foundations, "Insight" includes measuring CMDB health in ways that are actionable-so teams can identify where data is degrading and prioritize remediation. Stale CIs (records that are no longer current, not updated, or no longer exist in the environment) are a common source of CMDB risk: they distort reporting, mislead impact analysis, and create noise for operational processes. Reducing staleness supports higher trust in the CMDB and better decision-making.
On the CMDB Health Dashboard, stale CI concerns align most closely with Compliance because staleness is fundamentally about whether CIs continue to meet defined governance expectations over time-such as update frequency, lifecycle status controls, and adherence to rules that indicate whether records are still valid.
Compliance scorecards typically highlight rule-based adherence to standards (for example, policies or health rules indicating that certain CI types must be recently updated or must not exceed staleness thresholds). This makes Compliance the right place to monitor and improve staleness.
By contrast, Completeness focuses on whether required attributes are populated (for example, owner, environment, support group). Correctness focuses on whether values are accurate and relationships make sense. While stale records can indirectly affect correctness, the dashboard tracking of "stale" is primarily treated as a governance adherence issue-whether the CMDB is being maintained according to lifecycle and freshness expectations-therefore it is surfaced through the Compliance scorecard.

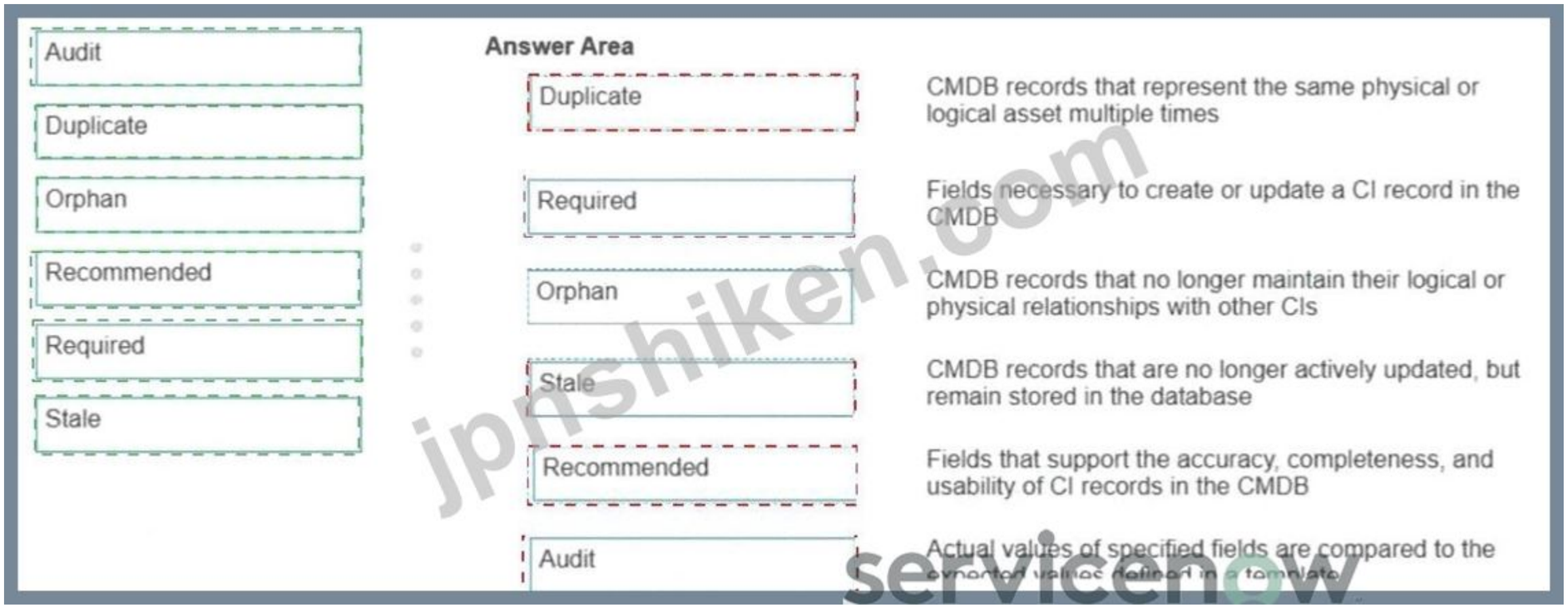
質問: **22**
プラットフォーム データ所有者は、5 つの検出ソースにわたる調整ルールを使用してデータ品質を向上したいと考えています。
データ所有者は、検出ソースの管理と監視にCMDB 360 / Multisource CMDBを含めることが最善の選択肢であることを認識しています。現在、当社はCMDBに必要なITOM Discoveryライセンスを保有していません。
360 / マルチソース CMDB。
この場合、データ所有者は何ができるでしょうか?
A. マルチソースIREルールを利用するには、ITOM Discoveryを購入する必要があります。

B. IRE調整ルールはCMDB 360が有効かどうかに関係なく検出ソースを使用できます。
C. CMDB 360 / Multisourceは、すぐに使用できるプラットフォーム製品です。
正解: **(**正解を表示します**)**
The Identification and Reconciliation Engine (IRE) is a core platform capability in ServiceNow and does not require CMDB 360 / Multisource CMDB to function. Even without the ITOM Discovery license, organizations can still define and use IRE reconciliation rules across multiple data sources.
IRE rules are source-aware and can evaluate attributes based on source precedence, regardless of whether CMDB 360 is enabled. CMDB 360 enhances visibility, governance, and monitoring of multiple sources, but it is not a prerequisite for reconciliation logic itself.
Option A is incorrect because purchasing ITOM Discovery is not mandatory to use multisource reconciliation. Option C is also incorrect because CMDB 360 / Multisource CMDB is a licensed add-on, not a universally available platform feature.
Therefore, the Data Owner can proceed by configuring IRE reconciliation rules directly, making Option B the correct answer.

質問: **23**
CMDBヘルスダッシュボードのメトリックを説明欄にドラッグアンドドロップしてください。



正解:



Explanation:

CMDB records that represent the same physical or logical asset multiple times	Duplicate
Fields necessary to create or update a CI record in the CMDB	Required
CMDB records that no longer maintain their logical or physical relationships with other CIs	Orphan
CMDB records that are no longer actively updated, but remain stored in the database	Stale
Fields that support the accuracy, completeness, and usability of CI records in the CMDB	Recommended
Actual values of specified fields are compared to the expected values defined in a template	Audit

* Duplicate identifies data quality issues where the same asset is modeled more than once, violating CMDB trust principles.
* Required ensures minimum viable CI creation and update-critical for ingestion and lifecycle integrity.
* Orphan highlights broken relationships, which directly impact impact analysis, Service Mapping, and Change Management.
* Stale flags CIs that are no longer being refreshed by Discovery or other sources but still exist in the CMDB.
* Recommended improves completeness and usability, supporting operational and reporting use cases without being mandatory.
* Audit validates data correctness by comparing CI field values against predefined templates or policies.

This exact mapping is what ServiceNow expects in CMDB Health Dashboard-related exam questions and real-world implementations.

質問: **24**
CMDB 360 / マルチソースCMDBでは、動的照合ルールが有効になっています。管理要件に基づき、CMDB管理者は複数の動的照合ルールを設定する必要があります。
調整ルールの作成ウィザード内で使用できる動的ルール タイプはどれですか? (2 つのオプションを選択)
A. 最小値
B. 最も報告されたもの
C. 最終更新日
D. 最終作成日
正解: **B,C (**コメントを発表する**)**
CMDB 360 / Multisource CMDB extends the standard IRE by enabling Dynamic Reconciliation Rules, which determine attribute values dynamically based on incoming data patterns rather than fixed source priority.
Within the Create Reconciliation Rule wizard, two supported dynamic rule types are:
Most Reported (Option B): selects the attribute value that is reported most frequently across all sources. This is useful when multiple sources contribute data and consensus is a strong indicator of correctness.
Last Updated (Option C): selects the most recently updated value, which is useful for rapidly changing attributes such as IP address or operational state.
Option A (Smallest Value) and Option D (Last Created) are not supported dynamic reconciliation rule types in ServiceNow.
Dynamic reconciliation rules are particularly valuable in complex, multisource environments where rigid source precedence is insufficient and data confidence must be inferred.
Therefore, the correct answers are B - Most Reported and C - Last Updated.

質問: **25**
CMDB 構成マネージャーは、CMDB 内の 60,000 台のサーバーで構成されるサーバー クラスの CMDB ヘルス ダッシュボード - 正確性スコアカードのメトリックを確認しています。
重複メトリックの場合、正常な CI / 評価済み = 59,000 / 60,000 と表示されます。孤立メトリックの場合、正常な CI / 評価済み = 45,000 / 50,000 と表示されます。2 つのメトリック間で評価されるサーバー CI の範囲 (60,000 と 50,000) の違いを説明する構成はどれですか。
A. 孤立メトリックには、サーバークラスに構成されたCMDBグループがあります。
B. 孤立メトリックには、サーバークラスにヘルス包含ルールが設定されています。
C. 重複メトリックには、サーバークラスにヘルス包含ルールが設定されています。
D. 重複メトリックには、サーバークラスに構成されたCMDBグループがあります。
正解: **B (**コメントを発表する**)**
In ServiceNow, each CMDB Health metric can independently define which CIs are in scope for evaluation.
This scoping is controlled primarily through Health Inclusion Rules, not CMDB Groups.
In this scenario, the Duplicate metric evaluates all 60,000 Server CIs, indicating no inclusion rule is restricting its scope. In contrast, the Orphan metric evaluates only 50,000 Server CIs, which means 10,000 servers are intentionally excluded from that metric's evaluation.
This difference is explained by a Health Inclusion rule configured specifically for the Orphan metric on the Server class. Health Inclusion rules allow administrators to define conditions-such as lifecycle state, environment, discovery source, or operational status-that determine whether a CI should be included in a specific health calculation. For example, retired servers or servers in build states may be excluded from orphan checks.
CMDB Groups are not used by the CMDB Health Engine to determine metric scope; they are used for reporting, assignment, and operational grouping. Therefore, Options A and D are incorrect.
Option C is also incorrect because the Duplicate metric clearly evaluates the full population of 60,000 servers.
Thus, the scope difference is correctly explained by the Orphan metric having a Health Inclusion rule configured, making Option B the verified answer.

質問: **26**

CMDB 構成管理チームは、識別および調整エンジン (IRE) を介して CMDB に取り込まれたデータから生成された重複排除タスクを管理したいと考えています。
これらの重複排除タスクは、CMDB ワークスペースのどの領域にありますか?
A. ホームタブの下にある重要なアクションタイル
B. Insights タブの CMDB 機能導入タイル
C. 「自分の作業」タブの合計ステータス
正解: (正解を表示します)
In ServiceNow, de-duplication tasks generated by the Identification and Reconciliation Engine (IRE) are operational governance tasks that require action by CMDB administrators or data owners. These tasks are surfaced in a role-based and actionable manner to ensure they are addressed promptly.
Within the CMDB Workspace, such tasks are accessed via the My Work tab, specifically under the Total status section. This area consolidates all actionable CMDB-related work items assigned to the user or their groups, including duplicate CI remediation tasks, data certification tasks, attestation requests, and lifecycle governance actions generated by CMDB Data Manager and IRE processes.
Option A is incorrect because the Home tab focuses on high-level navigation and featured actions, not task execution. Option B is also incorrect because the Insights tab and feature adoption tiles provide visibility and analytics, not task management.
By centralizing de-duplication tasks in My Work, ServiceNow ensures CMDB governance is embedded into daily operations, improving responsiveness and data quality while maintaining accountability.
Therefore, the correct answer is C - Total status under the My Work tab.

質問: **27**
CMDB管理者は、外部データをCMDBにインポートする必要があります。重複データの作成リスクを軽減し、不正なソースからの更新を防ぐため、識別 ·調整エンジン(IRE)がバイパスされないようにする必要があります。
識別および調整 API を使用して CMDB にデータをインポートするための推奨方法は何ですか?
A. インテグレーションハブ ETL
B. テーブル API (REST API または SOAP API)
C. インポートセットと変換マップ
正解: (正解を表示します)
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.

質問: **28**
CMDBヘルスダッシュボードは、正確性、コンプライアンス、完全性という3つの主要業績評価指標(KPI)に基づいています。各KPIには、複数のサブメトリクスが含まれています。
サブメトリクスをKPIにドラッグしてください。一部のオプションは適用されない場合があります。

正解:

Explanation:
Completeness # Required
Compliance # Audit
Correctness # Orphan
The CMDB Health Dashboard helps administrators measure whether the CMDB is trustworthy, usable, and aligned with governance expectations. The three main CMDB Health KPIs are Completeness , Compliance , and Correctness , and each KPI evaluates a different aspect of CMDB data quality.
Completeness focuses on whether required or expected fields are populated on CI records. A CI may exist in the CMDB, but if important fields such as owner, support group, location, serial number, or environment are missing, the record is not complete enough to support operational processes. Therefore, Required belongs under Completeness .
Compliance focuses on whether CIs follow defined rules, policies, and governance requirements. The Audit sub-metric is associated with Compliance because audits check whether records meet expected standards and whether governance controls are being followed.
Correctness focuses on whether the CI data is accurate and logically valid. Orphan belongs under Correctness because an orphan CI lacks expected relationships or context. For example, a server without expected dependencies or service relationships may reduce the accuracy of impact analysis and operational reporting.
The other visible options, such as Stability , Suggested , and Certify , are not the best fit for the three answer slots shown in this question.

質問: **29**
(2つ選択してください)
CMDB管理者は、CMDBデータ基盤ダッシュボードに表示される指標を監視するかどうかを評価している。
このダッシュボードで結果を継続的に監視するという決定を裏付けるメリットは何ですか？
A. 健康監査に不合格となったすべてのCIのリストを提供します
B. 過去90日間に更新されたアクティブなCIに関する指標を提供します。
C. 識別 調整エンジン(IRE)によって処理された構成アイテム(CI)のメトリクスを提供します。
D. CMDB内のすべての孤立した構成アイテムに関するレポート
正解: **(**正解を表示します**)**

The CMDB Data Foundations Dashboard in ServiceNow is intended to provide ongoing, trend-based visibility into how well the CMDB ingestion and maintenance processes are functioning-not just point-in- time issue lists. This is why continual monitoring of its metrics is valuable.
Option B is correct because tracking active CIs updated in the last 90 days provides a strong indicator of data freshness and operational relevance. A healthy CMDB should reflect recent updates from Discovery, integrations, and governed manual processes. Monitoring this metric over time helps organizations detect stagnation, discovery failures, or integration issues early.
Option C is also correct because metrics for CIs processed by the Identification and Reconciliation Engine (IRE) directly indicate the effectiveness and adoption of governed ingestion practices. Consistent IRE processing confirms that integrations are not bypassing identification rules, reducing duplicates and improving trust in CMDB data. Trending this metric helps validate Data Foundations maturity.
Option A is incorrect because the dashboard is not designed to provide exhaustive audit failure lists; those are handled through certification and remediation workflows. Option D is also incorrect because orphan CIs are a specific health condition surfaced via health rules and remediation tools, not a core benefit metric for continual dashboard monitoring.
Therefore, the correct answers are B and C.

質問**: 30**
ServiceNow の CMDB Data Manager の目的または要件は何ですか?
A. アーカイブされたレコードを暗号化してセキュリティを強化します
B. CMDB内のCI間の関係ルールの適用を自動化します。
C. 保持ポリシーに基づいてレコードのアーカイブと削除を自動化します。
正解**: (**正解を表示します**)**
The CMDB Data Manager capability in ServiceNow is designed to support CMDB governance, specifically around data lifecycle management. Its primary purpose is to ensure that CI records are retained, archived, and deleted in accordance with defined retention policies, regulatory requirements, and organizational data governance standards.
As CMDBs mature, they naturally accumulate obsolete, retired, or decommissioned CIs. If these records are not properly managed, they negatively impact CMDB health, reporting accuracy, discovery reconciliation, and performance. CMDB Data Manager addresses this by automating the archival and deletion of records once lifecycle conditions and retention thresholds are met.
Option A is incorrect because encryption of archived records is handled by platform-level security and data protection features, not CMDB Data Manager. Option B is also incorrect because relationship rule enforcement is managed through CSDM guidance, CMDB relationship rules, and identification/reconciliation logic-not by CMDB Data Manager.
By automating retention-based archival and cleanup, CMDB Data Manager helps organizations maintain a lean, compliant, and high-quality CMDB, which directly supports CMDB Health metrics such as correctness and compliance.
Therefore, the correct and verified answer is Option C.

質問**: 31**
金融サービス企業のエンタープライズアーキテクトが企業全体で業務に携わり、各部門の能力を追跡したいと考えています。CSDM 5のどのドメインが使用されていますか？(1つ選択してください)
A. 財団
B. ビルドと統合 (ビルド)
C. サービス消費(販売/消費)
D. 設計と計画(デザイン)
E. サービス提供(技術管理)
正解**: D (**コメントを発表する**)**
In CSDM version 5, enterprise-wide capability tracking is firmly positioned within the Design and Planning domain. This domain is specifically intended to support Enterprise Architecture, Portfolio Management, and Strategic Planning use cases, making it the correct choice for an Enterprise Architect working across the organization.
The Design and Planning (Design) domain focuses on answering "what the business needs and how it should be designed" before services are built or delivered. It includes core concepts such as Business Capabilities, Value Streams, Information Objects, Business Applications, and Architectural relationships. Business capabilities represent what an organization does to achieve its objectives,

independent of organizational structure, technology, or implementation. This abstraction is essential for enterprise architects, especially in highly regulated industries like financial services, where strategic alignment and impact analysis are critical.
The other domains do not fit this requirement:
* Foundation provides shared reference data (companies, locations, users) but does not model enterprise capabilities.
* Build and Integration focuses on application development, CI/CD pipelines, and integration layers.
* Service Consumption is concerned with customers, offerings, and how services are consumed.
* Service Delivery models the operational and technical delivery of services, including infrastructure and runtime environments.
From a Data Foundations and CSDM governance perspective, tracking enterprise capabilities in the Design and Planning domain ensures:
* Clear separation between strategy, design, and operations
* Alignment with ITIL 4 strategy and planning practices
* Strong support for impact analysis, rationalization, and transformation initiatives Therefore, Design and Planning (D) is the correct and fully aligned CSDM 5 domain for tracking enterprise capabilities.

質問: 32
企業では、必要なCSDM成熟度レベルを理解する必要があります。様々なステークホルダーが、将来的に期待されるいくつかのユースケースを挙げています。
情報オブジェクトが必要なユースケースはどれですか?
A. 資産管理チームは、ビジネスアプリケーションのコンテキストにおける資産ライフサイクルのコンプライアンスを理解したいと考えています。
B. イベントオペレーションチームは、イベントをインシデントとして自動化し、運用上のアクションを実行したいと考えています。
C. カスタマーサービスチームは、プロアクティブなケース管理を導入したいと考えています。
D. SecOpsチームは、ビジネスアプリケーションのコンテキストにおける運用リスクを理解したいと考えています。
E. ビジネスサービス管理チームは、消費者側への運用上の影響を把握したいと考えています。
正解: **(正解を表示します)**
Within the Common Service Data Model (CSDM), information objects are used to represent non-CI data entities that provide important business or governance context but are not configuration items themselves.
These objects are especially important when extending service visibility beyond pure infrastructure and application relationships.
The use case described in Option A-understanding asset lifecycle compliance in a Business Application context-explicitly requires information objects. Asset lifecycle data (such as financial state, depreciation, warranty, and compliance milestones) is typically managed in IT Asset Management (ITAM) and must be associated to Business Applications without converting every asset-related data point into a CI. Information objects enable this linkage while maintaining clean CMDB boundaries.
Option B focuses on event-to-incident automation, which relies on CIs, technical services, and operational relationships, not information objects. Option C (proactive case management) is primarily a CSM and service offering use case. Option D (SecOps risk context) relies on application services and business application relationships, not information objects. Option E (business service impact) is addressed through service modeling and service mapping, again without requiring information objects.
Information objects are introduced as organizations mature and need to integrate governance, financial, or compliance data with service and application models-making asset lifecycle compliance the correct match.
Therefore, the correct answer is A.

質問: **33**
CMDB 管理者は、CMDB データ基盤ダッシュボードを表示して、一意の場所の結果の割合が低いことに気付きました。
この問題を修正するために、関連するプレイブックから推奨されるプロセスは何ですか?
A. 重複CI修復機能を使用して重複した場所レコードをマージします。
B. 組織の標準命名規則に一致する場所を保持し、重複した場所をそれ以上検証せずに削除します。
C. 両方の場所を確認し、正しい場所でCIを更新し、重複した場所を削除します。
D. どちらも有効な代替場所として使用できるため、両方の場所を保持します。
正解: **(**正解を表示します**)**
In Data Foundations, a low "Unique Locations" result indicates the CMDB contains duplicate Location records (or inconsistent location usage) that reduce data reliability for operational reporting and service analytics. This sits under "Insight" because the dashboard is highlighting a measurable quality issue, and the playbook provides a structured remediation path to restore confidence.
The recommended playbook-aligned process is to review the duplicate locations, determine which one is correct per the organization's naming and data standards, then re-point dependent records (CIs) to the correct Location before removing the duplicate. That sequence is essential: if you delete a duplicate location without first updating related CIs, you risk leaving CIs with missing/invalid location references, creating new completeness and correctness issues.
Option C matches best practice: validate which location record should remain, update all affected CIs (and any other dependent records) to use the correct location, and then delete (or retire/merge, depending on process controls) the duplicate record. Option B is risky because it skips validation and dependency updates.
Option D defeats the purpose of "Unique Locations" by allowing ongoing inconsistency. Option A is not the best fit because "Duplicate CI Remediator" is oriented toward CI duplicates; Location is typically a reference
/organization data domain and is handled through the specific remediation steps for that data set. Therefore, C is the recommended process.

質問: **34**
CMDB 360 / マルチソース CMDB 保存クエリは、CMDB ワークスペースのどこで表示および作成できますか?
A. CMDB 360タブのカバレッジウィンドウ
B. Insightsタブの保存されたクエリウィンドウ
C. CMDBクエリビルダー
D. CMDB 360 タブの保存済みクエリウィンドウ
正解: **D (**コメントを発表する**)**
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.

質問: **35**
(2つ選択してください)

構成管理プロセスのオーナーは、カスタム構成アイテムをCMDBに取り込むためのソリューション案を、技術ガバナンス部門へのプレゼンテーション用に準備しています。このソリューションは、ベストプラクティスに準拠し、将来の技術的負債を最小限に抑え、アップグレードのコンプライアンスを確保する必要があります。
これを実現するソリューションはどれですか？
A. 基本となるCIクラスを再利用し、必要に応じて属性名を変更する
B. CMDB CIクラスモデルストアアプリケーションをインストールまたはアップグレードして、新しい属性に対応する適切な既存のCIクラスを見つける
C. 既存のアセットクラステーブルを拡張して、カスタムCIクラス属性を追加する
D. 既存のCIクラステーブルを拡張して、カスタムCIクラス属性に対応する
正解: **(**正解を表示します**)**
In ServiceNow, ingesting custom CIs must be done with a strong focus on upgrade safety, governance, and long-term maintainability. Data Foundations guidance explicitly discourages repurposing or overloading base classes, as this creates technical debt and upgrade risk.
Option B is a best practice because the CMDB CI Class Models Store delivers ServiceNow-supported CI classes that align with platform evolution. Before creating or extending classes, administrators should verify whether a suitable class already exists or has been introduced in newer releases. This avoids duplication and ensures future compatibility.
Option D is also correct. When no suitable class exists, extending an existing CI class (under the appropriate parent) to add required attributes preserves inheritance, discovery behavior, reporting, and upgrade compatibility. This approach is preferred over creating entirely new, disconnected schemas.
Option A is incorrect because repurposing base classes and renaming attributes breaks standard semantics, causes confusion, and complicates upgrades. Option C is incorrect because extending Asset tables to represent CIs conflates ITAM and CMDB concerns; assets and CIs serve different purposes and lifecycles.
Therefore, the solutions that minimize technical debt and ensure upgrade compliance are B and D.

質問: **36**
(2つ選択してください)
CMDB管理者は、シアトルにあるアプリケーションサービスに接続されているすべてのデータベースを検索するCMDBクエリを作成したいと考えています。また、それらのデータベースに関連するインシデントもクエリに含めたいと考えています。
このクエリを作成するために、どのような手順を踏むべきですか？
A. 非CMDBテーブルリストからインシデントテーブルをキャンバスに追加します。
B. アプリケーションサービスノードにプロパティ列を追加します
C. データベースノードにロケーション＝シアトルのフィルタを追加します。
D. 関係レベルを最大2レベルの関係に設定します
正解: **A,C (**コメントを発表する**)**
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).

質問: **37**
ACMDB 管理者は、重複する CI の対処と修復を防止するために使用できるツールを理解しようとします。各機能を対応する結果とドラッグ アンド ドロップします。一部のオプションは適用されない場合があります。

正解:

Explanation:

Feature	Corresponding Outcome
Certification Tasks	Can be assigned to groups for resolving duplicate CIs
CMDB Health Dashboard – Correctness Scorecard	Offers insight into duplicate CIs within the CMDB
De-duplication Templates	Offers a solution to resolve de-duplication tasks in bulk
Duplicate CI Remediator	Provides a wizard to resolve de-duplication tasks individually

In ServiceNow, duplicate CI management spans detection, insight, and remediation:
The CMDB Health Dashboard - Correctness surfaces where duplicates exist and their impact, providing visibility and prioritization.
Certification Tasks support governed resolution, allowing assignment to individuals or groups to validate and correct data.
De-duplication Templates enable bulk remediation, ideal when standardized merge rules can be applied across many duplicates.
The Duplicate CI Remediator offers a guided, record-by-record wizard, preserving relationships and history for safer individual resolutions.
De-duplication Tasks are the work items generated by these processes, but they do not themselves represent an outcome-hence not applicable .
This mapping aligns with CMDB Data Foundations best practices, ensuring duplicates are identified, prioritized, and remediated efficiently and safely.

質問: 38
資産管理者は、資産レコードと CI レコードが自動的に同期された状態に保たれるようにしたいと考えています。
マネージャーはこれをどのように行いますか? (2 つ選択してください)

A. 資産テーブルの変更時にCIフィールドを更新するビジネスルールがアクティブであることを確認します。
B. CIテーブルの変更時に資産フィールドを更新するビジネスルールがアクティブであることを確認します。
C. 同期を確実に行うために、スケジュールされたジョブが営業時間外に実行されるようにしてください。
D. アセットとCI間の1対1の物理マッピングを確保する
正解**: A,B (**コメントを発表する**)**
In Data Foundations, "Govern" includes maintaining consistency between closely related data domains so that operational and asset lifecycle processes remain trustworthy. For hardware endpoints (and other asset-backed items), Asset and CI records often represent the same real-world item from two perspectives: financial /lifecycle management (Asset) and operational/service context (CI). Automatic synchronization prevents drift such as different owners, locations, or statuses appearing in Asset vs CI, which can cause reporting errors, incorrect fulfillment decisions, and audit issues.
The platform supports synchronization through out-of-box synchronization logic (commonly implemented as business rules and related configuration) that updates the CI when an Asset changes, and updates the Asset when a CI changes-so the records stay aligned regardless of which team updates which record. Therefore, ensuring the Asset-to-CI update logic is active (A) and the CI-to-Asset update logic is active (B) is the right way to keep data synchronized automatically.
Option C is not required for synchronization; the intent is near real-time consistency driven by record changes, not batch jobs that run later. Scheduled jobs might exist for other maintenance activities, but they are not the primary mechanism for keeping Asset/CI synchronized. Option D is important as a data quality principle-ideally one Asset maps to one CI for physical items-but "ensuring one-to-one mapping" is not the automation mechanism itself. Automation is achieved through the active synchronization logic, which is why A and B are the correct answers.

質問**: 39**
CMDB管理者は、経営陣へのプレゼンテーションに必要な情報を収集する任務を負っています。管理者は、重複CI、孤立CI、および古くなったCIの指標を提供する必要があります。
CMDB ヘルス ダッシュボードでこの情報を提供するスコアカードはどれですか?
A. 正確性
B. コンプライアンス
C. 完全性
正解**: B (**コメントを発表する**)**
In Data Foundations, "Insight" is about using health metrics to understand whether the CMDB is trustworthy and operationally usable. The CMDB Health Dashboard organizes health into scorecards that group related checks. When leadership asks for Duplicate CIs, Orphan CIs, and Stale CIs, they are asking about the CMDB' s ongoing hygiene and adherence to governance expectations-not simply whether fields are filled in.
These three metrics align best with Compliance because they represent whether CI records comply with rules that keep the CMDB clean and current over time:
* Duplicate CIs indicate the identification and governance controls are not consistently preventing multiple records for the same real-world item.
* Orphan CIs indicate relationship governance gaps-records exist without expected relationships or context, reducing their operational value and often signaling incomplete ingestion or lifecycle control issues.
* Stale CIs indicate lifecycle management gaps-records are not being updated, attested, retired, or archived appropriately, which increases risk and noise in operational processes.
By contrast, Completeness focuses on whether required attributes are populated (owners, support group, environment, etc.). Correctness focuses on whether values and relationships are accurate and logically valid.
While duplicates/orphans/stale can affect correctness indirectly, the dashboard's grouping for these hygiene and lifecycle-oriented measures is best represented by Compliance, which is why B is the correct answer.

質問**: 40**
CMDB 構成マネージャは、CI クラス マネージャを使用して CI クラスのグループ所有権を定義し、CI クラス マネージャで指定された所有権の値を活用する必要があります。
CMDB データ マネージャ ポリシーを作成するときに、どのグループ参照フィールドを設定する必要がありますか?
A. 承認グループ

B. グループによって管理
C. サポートグループ
D. グループの変更
正解**: B (**コメントを発表する**)**
In ServiceNow CMDB governance, the CI Class Manager allows administrators to assign ownership and accountability at the class level. This ownership is used by governance tools-especially CMDB Data Manager-to automatically determine who receives tasks and actions related to data lifecycle management.
The group reference field that aligns with class ownership and is consumed by CMDB Data Manager policies is the Managed By Group. This field represents the team responsible for the technical stewardship and lifecycle management of CIs within that class.
When CMDB Data Manager executes policies such as retention, archival, or cleanup, it uses the Managed By Group to assign tasks to the appropriate data owners. This ensures governance actions are routed to the correct accountable team without manual intervention.
Approval Group, Support Group, and Change Group serve different purposes. Approval Group is used for workflow approvals, Support Group is used for operational ticket routing, and Change Group supports Change Management governance. None of these reflect data ownership at the class level.
Therefore, to leverage CI class ownership defined in the CI Class Manager within CMDB Data Manager policies, the correct field is Managed By Group, making Option B the verified answer.

質問**: 41**
CMDB管理者は、CMDBデータ基盤ダッシュボードを使用してCMDBに関する洞察を得ます。
結果は、いくつかの指標において低いスコアを示している。
CMDB管理者は、CMDBの健全性を向上させるためにどのような対策を講じることができますか？(2つ選択してください。)
A. 重要度と優先度の高い指標に焦点を当てる
B. 除外ルールを用いて指標を調整し、スコアを向上させる。
C. 各メトリックの横にリンクされている修復プレイブックを使用してください
D. 非稼働および廃止された構成アイテムを削除する
正解**: (**正解を表示します**)**
The CMDB Data Foundations Dashboard is designed to help administrators identify priority data issues and take structured remediation actions. When several metrics have low scores, the correct approach is not to randomly clean records or manipulate scoring rules. Instead, the administrator should prioritize the most important issues and use the recommended remediation guidance.
Option A is correct because Data Foundations emphasizes prioritizing metrics with Critical and High priority first. These metrics usually represent the most important issues affecting CMDB trust, operational readiness, service visibility, and downstream ITSM processes such as Incident, Change, and Problem Management.
Focusing on the highest-priority metrics ensures remediation effort is aligned with business and operational risk.
Option C is also correct because the dashboard provides Remediation Playbooks linked to specific metrics.
These playbooks give structured guidance for analyzing the issue, identifying impacted records, correcting the data, and improving governance so the problem does not return. This is the preferred Data Foundations approach because it supports repeatable, auditable, and best-practice remediation.
Option B is not recommended because changing exclusions simply to improve the score can hide poor data quality rather than fix it. Exclusion rules should only be used when records are legitimately out of scope.
Option D may be valid in some lifecycle cleanup scenarios, but blindly removing non-operational or retired CIs is not the primary recommended action for improving dashboard health.

質問**: 42**
CMDB 管理者は脆弱性対応またはセキュリティ インシデント対応を実装しており、リスクを推定し、タスクの優先順位を設定するために十分なコンテキストを顧客が持っていることを確認する必要があります。
CSDM データ基盤ダッシュボードのどの Get Well Playbook がこれに役立ちますか?
A. 親ロケーションのないロケーション

B. ビジネスアプリケーション関係を持つアプリケーションサービス
C. プロダクトオーナーのいない名前付きプロダクトモデル
D. CIライフサイクルステージのカスタムステータス値の割合
正解: **(**正解を表示します**)**
In ServiceNow, Vulnerability Response and Security Incident Response rely heavily on business context to accurately assess risk, prioritize remediation tasks, and communicate impact to stakeholders. From a CSDM (Common Service Data Model) perspective, this context is primarily delivered through properly modeled relationships between Application Services and Business Applications.
The "Application Services with Business Application Relationships" Get Well Playbook directly addresses this requirement. In CSDM, Application Services represent the technical, deployable services that run in the environment, while Business Applications represent the logical applications that support business capabilities.
When these two are correctly related, security teams can clearly understand which business processes, customers, and revenue streams are affected by a vulnerability or security incident.
Without this relationship, vulnerabilities may still be detected, but they lack meaningful prioritization. For example, a critical vulnerability on an application service supporting a revenue-generating or customer-facing business application should be addressed far more urgently than one tied to a low-impact internal tool. This relationship is what enables risk-based prioritization, rather than purely technical severity-based prioritization.
The other options do not fulfill this need. Location hierarchy issues (Option A) and CI lifecycle status consistency (Option D) relate more to CMDB hygiene and governance, not security context.
Product ownership gaps (Option C) affect accountability but do not directly enable risk estimation during security response.
Therefore, Option B is the correct and CSDM-aligned Get Well Playbook for ensuring sufficient business context in Vulnerability Response and Security Incident Response workflows.

質問: **43**
CMDB クラスの所有者は、CMDB ワークスペースの 『マイ ワーク』タブでタスクを受け取ります。
これらのタスクを生成しているのはどの CMDB 管理ツールですか?
A. 重複排除テンプレート
B. CMDBデータマネージャー
C. CMDBヘルスダッシュボード
正解: **B (**コメントを発表する**)**
The CMDB Data Manager is the ServiceNow capability responsible for generating actionable governance tasks and assigning them to CI class owners and data stewards. These tasks appear directly in the "My Work" tab within the CMDB Workspace, enabling proactive and role-based CMDB governance.
CMDB Data Manager focuses on data lifecycle management, including archival, retirement, and cleanup of CIs based on defined policies. When lifecycle rules or retention thresholds are met-or when human validation is required-the Data Manager creates tasks to prompt responsible owners to take action. This ensures that CMDB data remains accurate, compliant, and lean over time.
The CMDB Health Dashboard (Option C) provides visibility into health metrics such as completeness, correctness, and compliance, but it does not generate tasks. Similarly, De-duplication templates (Option A) support duplicate identification and remediation workflows, but they do not create ongoing governance tasks in the CMDB Workspace.
By surfacing tasks in "My Work," CMDB Data Manager operationalizes governance and embeds accountability into daily workflows, which is a key principle of CMDB Data Foundations.
Therefore, the correct answer is Option B - CMDB Data Manager.

質問: **44**
ServiceNowは、構成アイテム(CI)とリレーションシップを効果的に取り込み、管理、維持するために設計された、一連のCMDB管理ツールを提供します。
設計アーキテクチャを管理ツールにドラッグ＆ドロップしてください。(一部のオプションは適用されない場合があります。)

正解:

Explanation:
Agent Client Collector
Automated agent-based solution running patterns
Import Sets
Organization-built solution using transform maps
Service Graph Connector
Pre-built store integration solutions that require minimal customization ServiceNow Discovery Automated agentless solution running patterns Agent Client Collector: This is ServiceNow ' s agent-based approach. It requires a software agent to be installed on the target host to run discovery patterns locally.
Import Sets: This is the traditional, manual way to bring in data. You define the source, and the organization builds the logic (Transform Maps) to move that data into the CMDB.
Service Graph Connector: These are found in the ServiceNow Store. They are " certified " integrations built to work with specific third-party vendors (like SCCM or Jamf) with minimal setup.
ServiceNow Discovery: The classic agentless method. It probes the network and uses patterns to identify and map CIs without needing software installed on every machine.

質問: 45
顧客は、最近インポートされたサーバー レコードが ServiceNow Discovery によって検出された後、より具体的な CMDB クラスに自動的に再分類されることを希望しています。
検出プロセス中に、既存のサーバー レコードが Linux サーバー クラスと Windows サーバー クラスに再分類される場合、どの再分類操作が発生しますか?
A. クラススイッチ
B. クラスダウングレード
C. クラスアップグレード
正解: **(正解を表示します)**
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.

質問: **46**
CMDB 管理者は、インシデント フォームの CI 参照フィールドなど、CI 参照フィールドにプリンシパル クラスの CI のみが表示されるようにしたいと考えています。
CMDB 管理者はプリンシパル クラスをどこで指定しますか?
A. CMDBワークスペース
B. CI クラス マネージャー
C. システムプロパティ
D. CMDBデータマネージャー
正解: **B (**コメントを発表する**)**
Principal Classes are a key CMDB configuration concept used to control which CI classes are selectable in reference fields across ITSM processes. This helps reduce noise, prevent incorrect CI selection, and improve data quality.
In ServiceNow, Principal Classes are designated within the CI Class Manager. This tool allows CMDB administrators to manage the CI class hierarchy, define ownership, and explicitly mark classes as principal.
Once a class is marked as principal, its CIs become available in CI reference fields such as those on Incident, Change, and Problem forms.
Option A (CMDB Workspace) provides operational and analytical views but does not control schema-level class behavior. Option C (System Properties) does not manage class designation. Option D (CMDB Data Manager) governs lifecycle and data quality policies, not reference field behavior.
By configuring Principal Classes in the CI Class Manager, organizations ensure that only relevant, high-value CI classes are exposed to end users, aligning with Data Foundations best practices.
Therefore, the correct answer is B - CI Class Manager.

質問: **47**
CMDB で管理されている新しいアプリケーションを反映するには、新しいカスタム クラスが必要です。
このカスタム CI クラスを追加するために最低限必要なロールは何ですか?
A. sn_cmdb_admin と personalize_dictionary
B. sn_cmdb_admin と personalize_form
C. sn_cmdb_admin と personalize_dictionary
D. itil_admin と personalize_form
正解: **(**正解を表示します**)**
Creating a custom CI class in the CMDB is a dictionary-level configuration activity and must be performed with the correct minimum privileges to ensure governance and upgrade safety in ServiceNow.

The sn_cmdb_admin role is required because it grants administrative access to CMDB structures, including CI class hierarchy management. This role ensures that changes align with CMDB governance controls and Data Foundations practices.
The personalize_dictionary role is also required because adding a new CI class involves creating or extending dictionary entries (tables, attributes, inheritance). Without dictionary-level access, a user cannot define new classes or attributes in the CMDB schema.
Option A is incorrect because sn_cmdb_admin alone is insufficient without dictionary privileges. Option B and D focus on form personalization, which affects UI layout only and does not allow creation of new CI classes. Additionally, itil_admin is not the correct role for schema-level CMDB changes.
Therefore, the minimal and correct role combination is sn_cmdb_admin and personalize_dictionary, making Option C the verified answer.

質問: **48**
割り当てられた個人が CI レコードのフィールドを更新できるようにするタスクを作成するのは、どのタイプの CMDB データ マネージャ ポリシーですか。
A. 監査
B. 認定
C. 証明
D. コンプライアンス
正解: **(**正解を表示します**)**
In CMDB governance, different CMDB Data Manager policy types serve different validation and enforcement purposes. When the objective is to allow an assigned individual to review and update specific fields on a CI record, the correct policy type is Certification.
A Certification policy creates actionable tasks that require the assignee to validate and, if necessary, correct specific CI attributes, such as lifecycle status, support group, environment, or ownership. During certification, the user can directly update CI fields to bring the record into compliance with defined standards.
Attestation (Option C) only asks the user to confirm that a CI still exists or is still valid; it does not require or enable attribute-level updates. Audit (Option A) is used for reporting and evidence collection, not remediation. Compliance (Option D) measures adherence to rules but does not itself generate editable remediation tasks.
Certification is therefore the primary mechanism used when human validation and correction of CI data is required-making it a cornerstone of CMDB data quality management.
Hence, the correct answer is B - Certification.