complete your programming course

about resources, doubts and more!

Servicenow

(CIS-ITSM)

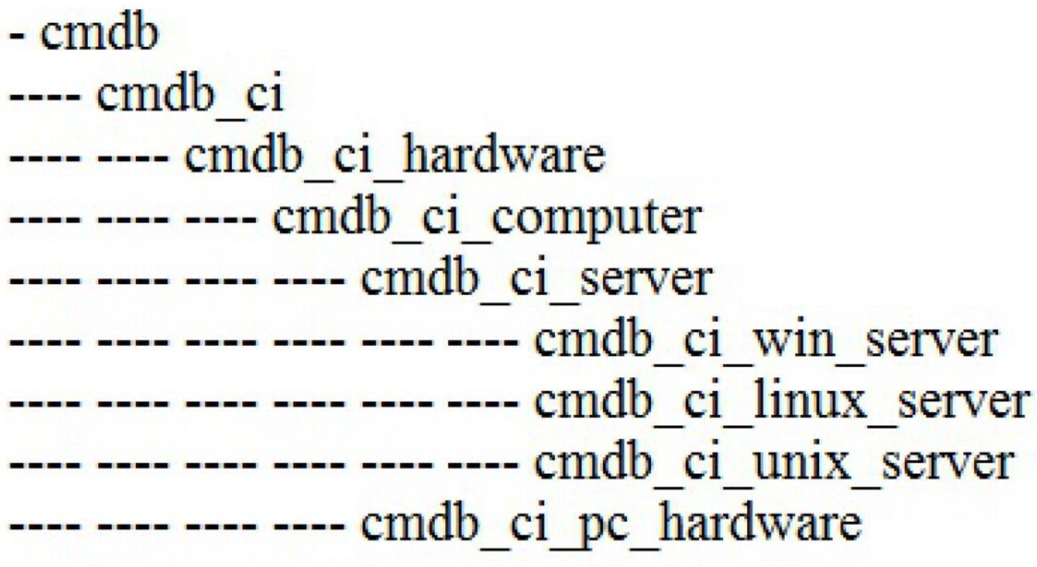
Certified Implementation Specialist - IT Service Management

Total: **224 Questions**

Link:

Question: 1

Given the class structure shown below, which types of CIs will be included in a report run against the cmdb_ci_computer table?



```
- cmdb
---- cmdb_ci
---- ---- cmdb_ci_hardware
---- ---- ---- cmdb_ci_computer
---- ---- ---- ---- cmdb_ci_server
---- ---- ---- ---- ---- ---- cmdb_ci_win_server
---- ---- ---- ---- ---- ---- cmdb_ci_linux_server
---- ---- ---- ---- ---- ---- cmdb_ci_unix_server
---- ---- ---- ---- cmdb_ci_pc_hardware
```

A.Just CIs defined directly in cmdb_ci_computer
B.CIs defined directly in cmdb_ci_computer and all parent classes C.CIs defined directly in cmdb_ci_computer and all child classes

Answer: C

Explanation:

CIs defined directly in cmd b_ ci_ computer and all child classes.

Question: 2

Which field from the configuration item will automatically populate in the Assignment group field of an incident record?

A.Managed by
B.Support group
C.Approval group
D.Change group

Answer: B

Explanation:

The correct answer is **B. Support group**. This is because ServiceNow is designed to automatically populate the Assignment group field on an Incident record based on the Support group specified for the Configuration Item (CI) affected by the incident. This automation aims to route incidents to the team best equipped to resolve issues related to that specific CI.

The Support group field on the CI record essentially designates the team responsible for maintaining and supporting that infrastructure component or service. When an incident is created referencing a particular CI, the system looks up the Support group defined for that CI and automatically populates the Assignment group field on the incident record. This streamlined assignment process minimizes manual intervention and ensures that incidents are quickly routed to the correct team, leading to faster resolution times.

Options A, C, and D (Managed by, Approval group, and Change group) are not directly related to the automated incident assignment process based on CI support responsibility. While these fields might be present on a CI record and are relevant in other workflows (such as change management or governance), they do not drive the default population of the Assignment group on an incident. The Managed by field typically refers to the business owner of the CI, while Approval group is used in approval workflows, and Change group is used in change management processes.

The Support group field is specifically created to designate the technical team responsible for support, directly linking the CI to the appropriate resolution team when an incident occurs. This facilitates efficient incident management and contributes to overall IT service quality.

For further research, refer to ServiceNow's official documentation on Configuration Management Database (CMDB) and Incident Management. Look into the relationships between CIs and assignment groups within the ServiceNow ecosystem.

ServiceNow Documentation (Search for "CMDB," "Incident Management," "CI Relationships," and "Assignment Rules")

Question: 3

Which of the following are defined for a given change model? (Choose three.)

A.Phase transitions

B.State model

C.State transition conditions

D.Phase model

E.State transitions

Answer: BCE

Explanation:

The correct answer of B. State model, C. State transition conditions, and E. State transitions aligns perfectly with the definition and functionality of change models within ServiceNow's ITSM module.

A change model serves as a reusable template, standardizing the change management process for specific types of changes. A state model (B) is fundamental as it defines the lifecycle of a change, encompassing various stages such as "New," "Assess," "Authorize," "Implement," "Review," and "Closed." This state model provides a clear and structured progression for each change request.

State transitions (E) outline the permissible movements between these states. For instance, a change request might transition from "Assess" to "Authorize" only after a thorough risk assessment. Each transition is governed by state transition conditions (C). These conditions act as gatekeepers, dictating the criteria that must be met before a change can move to the next stage. Examples of conditions include approvals being obtained, required fields being populated, or specific tasks being completed.

Therefore, change models meticulously define the state lifecycle, the allowable transitions between states, and the conditions that must be satisfied for each transition to occur. These features ensure consistency, predictability, and control within the change management process.

Options A and D are incorrect because while phases might be associated with changes, phase models and phase transitions aren't directly defined within a change model itself. The primary focus of a change model is its state-driven lifecycle. A phase model is more related to project management. Phase transitions describe the movement between phases within a project.

Further Research:

ServiceNow Documentation on Change Management: https://docs.servicenow.com/ (Search for "Change Management" and "Change Models")
ITIL 4 Foundation Handbook: Provides context on change enablement and models.

Question: 4

When is a change task for Post Implementation Review created for an unauthorized change?

A.When the change request moves to Close
B.When a change manager accepts the change
C.When the change request moves to a state of Review
D.When the change request moves to a state of Assess

Answer: C

Explanation:

The correct answer is C: When the change request moves to a state of Review. Here's why:

In ServiceNow's IT Service Management (ITSM) module, unauthorized changes represent deviations from established processes. A Post Implementation Review (PIR) is crucial to understand why the change was unauthorized and to prevent future occurrences. The PIR analyzes the effectiveness, risks, and lessons learned from the implementation, helping improve change management processes.

The PIR process often begins once the initial implementation phase has passed. Because the change was unauthorized, it is imperative to begin analysis of the situation quickly, which is ideally facilitated when the change moves to a state of "Review." The "Review" state signifies that the initial implementation is complete and ready for scrutiny.

Creating a PIR task when the change is in the "Review" state allows for a timely investigation into the unauthorized change. It provides an opportunity to capture relevant data and insights while the details of the change are still fresh.

Option A is incorrect because waiting until closure might delay the review process, potentially hindering the ability to gather complete information.

Option B is incorrect because Change Manager acceptance is not relevant for unauthorized changes as they bypass the normal approval flow.

Option D is incorrect because the "Assess" state is too early in the change process. The change needs to be implemented, even if unauthorized, before a meaningful review can occur. It would be illogical to review before any actions occur.

Therefore, creating a PIR task when the change moves to the "Review" state is the most appropriate time to initiate the investigation of an unauthorized change.

While the ServiceNow documentation doesn't specifically detail the PIR task creation trigger for unauthorized changes, it emphasizes the importance of PIRs in the overall change management lifecycle and their role in continuous improvement. You can find details about change management workflows, states, and Post-Implementation Reviews in the official ServiceNow documentation.

https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/change-management/concept/change-management-process.html

Question: 5

Which should be used to explore the entire hierarchy and table definitions of the Configuration Management Database Classes?

A.Reports
B.CI Class Manager
C.Application Menus
D.Dependency View

Answer: B

Explanation:

The correct answer is B, the CI Class Manager. Here's why:

The CI Class Manager is a dedicated ServiceNow application specifically designed for exploring and managing the Configuration Management Database (CMDB) classes. It provides a visual interface to navigate the entire hierarchy of CMDB classes, revealing their parent-child relationships. Through the CI Class Manager, you can view the definitions of each class, including the attributes (fields) they contain, and their relationships with other classes. This allows administrators and developers to understand the CMDB structure comprehensively.

Option A, Reports, is incorrect because while reports can display data from the CMDB, they don't offer a direct way to explore the underlying table definitions and hierarchical relationships.

Option C, Application Menus, is incorrect because application menus are simply navigation tools and do not provide insights into CMDB class structures.

Option D, Dependency View, focuses on visualizing relationships between CIs, not on exploring the class definitions and hierarchy itself. While useful for understanding dependencies, it doesn't provide the same level of detail about the classes as the CI Class Manager.

In essence, the CI Class Manager is the tool built specifically to allow users to navigate the CMDB, see the definitions of all the various classes that are available and their hierarchies.

Authoritative link:

ServiceNow Documentation: https://docs.servicenow.com/bundle/sandiego-servicenow-platform/page/product/configuration-management/concept/cmdb-class-manager.html

Question: 6

Which of the following cannot be defined or set through a Catalog UI Policy?

A.Setting a variable to mandatory
B.Apply a requirement to all form views
C.Setting a catalog category to visible
D.Setting a variable to read-only

Answer: B

Explanation:

The correct answer is **B. Apply a requirement to all form views**. Here's a detailed justification:

Catalog UI Policies are designed to dynamically control the behavior and appearance of items within the Service Catalog based on conditions. They operate on catalog items and variables within the catalog request. They allow you to make variables visible, mandatory, or read-only based on user input or other criteria on the catalog item form itself. Options A (Setting a variable to mandatory), C (Setting a catalog category to visible), and D (Setting a variable to read-only) are all functionalities directly supported by Catalog UI Policies. You can configure these policies to change the visibility, mandatory status, or read-only status of catalog variables based on specified conditions.

However, Catalog UI Policies are specifically designed for the Service Catalog and apply to the catalog item form presented to users before submitting a request. They don't directly control the behavior of the resulting request form or task forms, nor do they influence all form views across the entire platform. Attempting to apply a requirement to all form views would require scripting or UI Policies on the task or request tables themselves, not a catalog UI Policy. Furthermore, Catalog UI Policies are context-specific and operate within the context of the Service Catalog item. They do not have the scope to affect all form views globally. The form view configuration is typically managed at the table level or through global UI policies.

Therefore, while UI Policies at the table level (e.g., incident, change request) can affect form views, Catalog UI Policies are restricted to the catalog item form displayed during the request submission process.

Relevant documentation:

ServiceNow Docs: Catalog UI Policies
ServiceNow Docs: UI Policies (Distinguishes between UI Policies and Catalog UI Policies).

Question: 7

Which type of catalog item should be used to create an incident record from the portal?

A.Incident Template
B.Request Item
C.Order Guide
D.Record Producer

Answer: D

Explanation:

The correct answer is D, Record Producer. Here's a detailed justification:

Record Producers are specifically designed to create task-based records, such as Incidents, Problems, or Changes, directly from the service portal or catalog. They provide a user-friendly interface for submitting information required to generate these records. Unlike other catalog item types, record producers directly create the target record (e.g., Incident).

Incident Templates (A) are used to pre-populate fields within an existing incident record. While helpful, they don't initiate the incident creation process themselves from the portal. A user still needs to trigger the creation of the incident first, often manually.

Request Items (B) are used to fulfill requests for tangible goods or services. The primary goal is the delivery of an item or a service via a workflow, not the creation of an incident record for reporting an issue. Request Items do not create incidents directly. They may trigger incident creation as part of their fulfillment workflow, but they aren't designed for a user to directly generate an incident from the portal.

Order Guides (C) bundle multiple catalog items (including request items and potentially record producers) together into a single request. While an order guide could include a record producer for creating an incident, the order guide itself isn't the primary mechanism for creating incidents; it's the included record producer. The order guide's primary function is to guide users through a series of related requests.

Record producers, on the other hand, provide a streamlined and direct way for users to report incidents from the portal. They present a simple form where users can input relevant information. Upon submission, the record producer uses this information to automatically create a new incident record, routing it to the appropriate assignment groups and setting the initial state based on pre-configured values. They allow for scripting and advanced configurations tailored for incident creation. This makes them the ideal mechanism when the goal is to create an incident from the service portal.

In summary, Record Producers directly facilitate the creation of incident records from the service portal, providing a tailored user interface and automated record generation capabilities.For further research, consult the official ServiceNow documentation:

Record Producers:https://docs.servicenow.com/en-US/bundle/sandiego-it-service-management/page/service-catalog/concept/c_RecordProducers.html

Question: 8

Which incident management roles are activated by installing the ITSM Roles plugin (com.snc.itsm.roles)? (Choose two.)

A.sn_incident_read
B.itsm_incident_read
C.incident_manager
D.sn_incident_write
E.itsm_incident_write

Answer: AD

Explanation:

The ITSM Roles plugin (com.snc.itsm.roles) in ServiceNow activates several roles that control access and permissions related to IT Service Management functionalities. Specifically, regarding incident management, the plugin activates the sn_incident_read and sn_incident_write roles.

The sn_incident_read role grants users the ability to read and view incident records. This role is crucial for individuals who need to monitor incident progress, analyze trends, or investigate past incidents but don't require the ability to modify incident data.

The sn_incident_write role allows users to create, update, and resolve incident records. This role is essential for incident analysts, support staff, and others directly involved in the incident management process. Users with this role can document symptoms, assign incidents to appropriate groups, track progress, and ultimately resolve the issue.

The itsm_incident_read and itsm_incident_write roles are not activated directly by installing the com.snc.itsm.roles plugin; rather, the sn_incident_* roles are the ones tied to the plugin activation within the specific ServiceNow core incident management scope. It is important to differentiate between these roles as they have distinct origins and behaviours within the ServiceNow platform. The incident_manager role exists outside of the plugin activation in the base system as a role to delegate management of the entire incident process.

Therefore, the correct answer is A (sn_incident_read) and D (sn_incident_write) as these are the incident management roles activated by the ITSM Roles plugin (com.snc.itsm.roles). Other roles may be available for Incident Management. However, ITSM Roles plugins does not directly activate them.

ServiceNow Docs - ITSM Roles PluginServiceNow Docs - Base System Roles

Question: 9

A customer requests that when the Service Desk agent clicks on the information icon for the Caller's name, the quick view frame shows only the following fields:

User name -

Manager name -

Email Address -

Employee ID -

How would you modify the quick view frame?

A.Update the sys_popup view for the user table
B.Update the sys_quick view for the caller table
C.Update the sys_popup view for the caller table
D.Update the sys_quick view for the user table

Answer: A

Explanation:

The correct answer is **A. Update the sys_popup view for the user table.**

Here's a detailed justification:
The "quick view" (or "hover") functionality in ServiceNow, which displays information when hovering over a reference field's information icon, is controlled by views. When a user hovers over the Caller field (which is a reference to the User [sys_user] table), the system needs to determine which view to display. The sys_popup view is specifically designed for this purpose.

Here's why the other options are incorrect:

B. Update the sys_quick view for the caller table: There isn't a standard "sys_quick" view mechanism directly controlling hover information displays in ServiceNow. The sys_popup is the dedicated view for this.
C. Update the sys_popup view for the caller table: The Caller field references the User table, it doesn't have a table that governs the fields displayed within the popup on hover. The sys_user table defines the details of each user.

D. Update the sys_quick view for the user table: As stated before, sys_quick view is not related to hover information displays.**Detailed Explanation:**

1. **Reference Fields:** The Caller field on the Incident table is a reference field. It points to a record in another table, which is the User [sys_user] table.

2. **Hover Behavior:** When you hover over the information icon next to a reference field, ServiceNow displays a popup window. This window shows information related to the record being referenced.

3. **Views:** Views in ServiceNow are named arrangements of fields on a form or list. They determine what information is displayed to the user under certain circumstances.

4. **The sys_popup View:** The sys_popup view is a specialized view specifically designed for popup windows in reference fields. When ServiceNow displays the popup window for the Caller field, it looks for the sys_popup view defined on the User [sys_user] table. If it does not exist, the default view will show the full table.

5. **Customization:** To change the fields that appear in the quick view, you need to modify the sys_popup view on the referenced table (in this case, the User [sys_user] table). You would add the desired fields (User name, Manager name, Email Address, Employee ID) to this view.

Steps to Modify:

1. Navigate to **System UI > Views**.
2. Search for "sys_popup".
3. Select 'sys_user' for Table.
4. Add/remove fields from the form layout to show/hide fields.

Example: To show User Name, Email Address, Manager name, and Employee ID, you would add those fields to the sys_popup view on the User [sys_user] table.**Authoritative Links:**

ServiceNow Docs - Views:https://docs.servicenow.com/bundle/vancouver-platform-administration/page/administer/form-administration/concept/c_Views.html
ServiceNow Community - Views:https://community.servicenow.com/community?id=community_article&sys_id=d16ff9a8dbb6d0103daa1ea8c49619b3

By modifying the sys_popup view, the customer's request will be satisfied: only the specified fields will be displayed when hovering over the Caller's information icon.

Question: 10

Your customer has built a mature knowledge base, with articles targeted to internal audiences -which are technical. Other articles are written for end users, with simple instructions. From the Incident form, the agents would like to be able to identify which articles are visible to the callers What feature would you use, to satisfy this requirement?

A.Internal/External Highlighting
B.Search as User
C.Show User Viewable
D.User Only View

Answer: B

Explanation:

The correct answer is **B. Search as User**. Here's why:

The requirement is for agents to identify, from the Incident form, which knowledge articles are visible to the caller (the end-user). The most direct and efficient way to achieve this is by using the **"Search as User"** functionality. This feature allows an agent to impersonate the caller and perform a knowledge search as if they were the end-user. This means the search results will be filtered based on the caller's roles, user criteria, and any other access restrictions configured on the knowledge base. The agent will then see only the articles the caller would see.

"Internal/External Highlighting" (A) could involve visually distinguishing articles based on their audience, but this is usually a pre-existing article property, and doesn't dynamically adapt to the specific caller's permissions. It doesn't filter the search results based on the end user's access rights.

"Show User Viewable" (C) is too vague. While a generic "viewable" flag could exist, it doesn't guarantee the same level of granular access control that "Search as User" provides. It would likely need substantial customization, which is less efficient than using the out-of-the-box functionality.

"User Only View" (D) is unclear and isn't a standard ServiceNow feature specifically designed for this purpose within the incident form's knowledge search. It sounds like it might filter to show only user-created content (if it existed), which is completely unrelated to the requirement.

Therefore, "Search as User" is the most suitable because it provides the capability to accurately simulate the end-user's knowledge search experience directly from the Incident form. This functionality leverages the platform's built-in access control mechanisms related to Knowledge Management.

Further research:

ServiceNow Documentation on Impersonating a User:https://docs.servicenow.com/bundle/utopia-platform-administration/page/administer/navigation_and_ui/task/t_ImpersonateAUser.html
ServiceNow Knowledge Management Documentation:https://docs.servicenow.com/bundle/utopia-servicenow-platform/page/product/knowledge-management/concept/knowledge-management.html

Question: 11

Your customer is using the baseline Create Incident Catalog Item and would like to add a few additional input fields. How should you update the catalog item?

A.Edit in Catalog Item Designer
B.Edit in Item Designer
C.Edit in Catalog Builder
D.Edit in Form Designer

Answer: C

Explanation:

The provided answer, C (Edit in Catalog Builder), is the most appropriate method for adding new input fields to an existing catalog item like the "Create Incident" item in ServiceNow's IT Service Management module.

Catalog Builder is designed to provide a guided, user-friendly interface for creating and maintaining service catalog items. It simplifies the process of adding variables (input fields), defining workflows, and setting up pricing. It provides a centralized and streamlined approach.

Option A (Catalog Item Designer) is not the correct option because the catalog item designer has not been released yet.

Option B (Item Designer) is for creating new catalog items or editing existing items that are not part of the standard service catalog.

Option D (Form Designer) is mainly used to customize the form layout of tables, including incident forms themselves. While you could theoretically add fields to the Incident table through the Form Designer and expose them via the catalog item, this is a less direct and generally discouraged approach. Modifying the underlying table schema for catalog item-specific fields is not best practice. This would modify the incident form for all users, potentially impacting unrelated workflows.

Catalog Builder is the best solution because you are adding input variables to an existing catalog item and not changing the table or form layouts. You simply want to add variables to the Service Catalog item form.

In summary, when dealing with augmenting existing catalog items with additional input fields (variables), Catalog Builder offers the most efficient, intuitive, and ServiceNow-recommended method for achieving the desired outcome within the IT Service Management context.

References:

ServiceNow Documentation: Service Catalog Items: (https://docs.servicenow.com/en-US/bundle/quebec-it-service-management/page/product/service-catalog-management/concept/c_ServiceCatalogItems.html) ServiceNow Documentation: Using the Catalog Builder: (https://docs.servicenow.com/bundle/utah-platform-administration/page/build-cloud-based-apps/native-platform/service-catalog/concept/catalog-builder.html)

Question: 12

Your customer is complaining that Service Desk users keep accidentally assigning Incidents to the Network CAB, instead of Network Support You have confirmed that:
The Network Support group record has the Group types: Incident and Change
The Network CAB group record has the Group type: Change
What could you do on the incident form, for the Assignment Group field, to resolve this issue?

A.Add a UI action to hide the Network CAB group from the list

B.Add a UI action to provide an error message if the Network CAB group is selected

C.Add Dictionary Override to specify the Incident group Reference Qualifier

D.Modify the choice list to include only the appropriate group types

Answer: C

Explanation:

The correct answer is C: Add a Dictionary Override to specify the Incident group Reference Qualifier. Here's why:

The problem is that both the Network Support and Network CAB groups appear in the Assignment Group field on the Incident form, leading to misassignment. The key to solving this is to filter the available Assignment Groups specifically for Incidents.

Option A, hiding the Network CAB group with a UI Action, isn't ideal. The Network CAB group might be legitimately needed in other contexts, such as Change Management. Hiding it outright limits functionality elsewhere.

Option B, an error message upon selection, is also not user-friendly. It requires the user to make the mistake before being notified, leading to a less efficient workflow. Prevention is better than correction.

Option D, modifying the choice list, isn't the appropriate method. The list is dynamically generated based on the group table, and directly altering it would be problematic and not scalable.

A Dictionary Override, specifically on the Assignment Group field for the Incident table, allows you to modify the reference qualifier. This qualifier acts as a filter for the groups that appear in the assignment group field.
By setting the reference qualifier to only include groups with the "Incident" group type, you ensure that only relevant support groups like "Network Support" are displayed when assigning incidents. This directly addresses the core issue without impacting other modules or requiring user error correction. This is the cleanest, most targeted, and most scalable solution. The dictionary override provides a table-specific setting that will only apply to incidents, leaving other tables untouched.https://developer.servicenow.com/dev.do#!/learn/courses/kingston/scripting_kingston/scripting_admin/

platform-administration/page/script/client-scripts/concept/c_ReferenceQualifiers.html

Question: 13

Which Agent workspace feature gives agents automatic search results that show possible solutions for records they open?

A.Chat Bot
B.Related Search Results
C.Knowledge Bases
D.Intelligent Agent
E.Agent Assist

Answer: E

Explanation:

The correct answer is E, Agent Assist. Agent Assist within ServiceNow's Agent Workspace is designed to proactively provide agents with relevant information and solutions as they work on records. Its core function is to present a context-sensitive pane containing knowledge articles, catalog items, community discussions, and other relevant content based on the information in the record being viewed (e.g., incident, problem, change).

This automatic search functionality eliminates the need for agents to manually search for solutions, significantly reducing resolution times and improving agent efficiency. By analyzing keywords and categories within the record, Agent Assist aims to anticipate the agent's needs and surface potential solutions directly.

Option A, Chat Bot, while helpful in self-service scenarios, doesn't automatically provide search results based on open records within the Agent Workspace in the same manner. Chatbots typically require user interaction to trigger searches.

Option B, Related Search Results, is a general term but doesn't capture the proactive and automated nature of Agent Assist within the Workspace. Agent Assist proactively pushes contextually relevant information rather than merely providing a standard search feature.

Option C, Knowledge Bases, represent the repository of information, but by themselves, they do not provide the automatic and targeted search results. Agent Assist leverages Knowledge Bases (and other sources) as part of its solution delivery.

Option D, Intelligent Agent, is a broader concept that encompasses various AI-powered capabilities. While Agent Assist is powered by intelligence, "Intelligent Agent" is not the specific feature providing automatic search results within the Agent Workspace.

Agent Assist promotes first-call resolution, reduces training time for new agents, and enhances the overall customer experience by enabling agents to quickly access relevant information and resolve issues efficiently.

For further research, refer to ServiceNow's official documentation on Agent Assist: https://docs.servicenow.com/bundle/utah-now-intelligence/page/administer/agent-assist/concept/agent-assist-overview.html and https://www.servicenow.com/content/dam/servicenow/other-documents/service-brief/sb-agent-workspace.pdf

Question: 14

Which capability provides visibility to data joined between multiple tables?

A.Database Views

B.Metric Tables

C.Published Reports

D.Custom Tables

E.Breakdown Sources

Answer: A

Explanation:

The correct answer is **A. Database Views**.

Database views in ServiceNow (and in general database management) provide a way to join data from multiple tables and present it as a single virtual table. This is crucial for creating consolidated reports and gaining a comprehensive understanding of relationships between data across different entities. They don't store data themselves; they are constructed from the underlying tables based on defined relationships (joins). This allows users to query and report on data that spans multiple tables as if it were a single entity, greatly simplifying reporting and analysis. Using a view hides the complexity of the underlying table structures and the join operations from the end-user.

Option B, Metric Tables, are used for measuring and tracking the effectiveness of IT processes over time but don't provide visibility into joined data from various tables in the same way as database views. Metric tables record measurements like resolution time and cost.

Option C, Published Reports, while displaying data from multiple tables, relies on an underlying data source, which could be a database view, and doesn't inherently provide the join mechanism itself. Reports use data from a single table or a table joined using a database view.

Option D, Custom Tables, are used to store new data, not to view joined data from existing tables. They simply store data defined by the user.

Option E, Breakdown Sources, define how data in indicator sources will be broken down for performance analytics and do not directly combine data from different tables for general reporting purposes.

Therefore, database views are specifically designed to create a single, unified view of data from multiple tables, making them the correct solution for the described capability.

Refer to ServiceNow's documentation on database views for more detailed information:

ServiceNow Docs: Database Views

Question: 15

What tools are available to the assignee to help resolve an Incident? (Choose two.)

A.Knowledge Articles

B.Workarounds

C.CI Class Manager

D.Incident Overview Dashboard

E.Enterprise CMDB Dashboard

Answer: AB

Explanation:

The correct answer is **A. Knowledge Articles** and **B. Workarounds**. Here's why:

Knowledge Articles: These are documented solutions to common problems or recurring issues. When an Incident is assigned, the assignee can search for relevant knowledge articles to see if the problem has been previously resolved. Knowledge articles provide a readily available resource to quickly address known issues and standardize resolution processes. ServiceNow's Knowledge Management application directly integrates with Incident Management, facilitating this process.

Workarounds: A workaround provides a temporary solution to bypass an immediate problem, allowing the user to continue working until a permanent fix is implemented. Offering a workaround is a crucial step in incident resolution, especially when the root cause is unknown or the permanent fix requires significant time. Assignees can document and share workarounds through the incident record, improving user experience and minimizing disruption.

Why the other options are not the best choices:

C. CI Class Manager: CI Class Manager is primarily used for defining and managing the attributes of Configuration Items (CIs) within the CMDB. While a correctly configured CMDB is valuable for Incident Management (understanding dependencies etc), the CI Class Manager itself isn't a direct tool to resolve an individual incident.

D. Incident Overview Dashboard: Incident Overview Dashboards provide a high-level view of incident metrics, trends, and performance. They are valuable for managers and analysts to monitor the overall incident management process. But, they don't give direct tools or methods for solving an individual incident for the assignee working on the specific problem.

E. Enterprise CMDB Dashboard: The Enterprise CMDB Dashboard offers a broad overview of the Configuration Management Database (CMDB). Although understanding the relationships between CIs (which the CMDB dashboard helps with) can be useful in incident investigation, the dashboard itself doesn't offer tools to directly solve incidents. The assignee may indirectly use the information from the CMDB to understand dependencies or identify problematic CIs, but this is not a direct resolution tool.

In summary, knowledge articles offer previously documented solutions, and workarounds provide temporary solutions. Therefore, these are the tools most directly applicable to resolving an assigned incident.

Authoritative Links:

ServiceNow Documentation: https://docs.servicenow.com/
ServiceNow Knowledge Management: https://docs.servicenow.com/bundle/sandiego-servicenow-platform/page/product/knowledge-management/concept/knowledge-management.html ServiceNow
Incident Management: https://docs.servicenow.com/bundle/sandiego-it-service-management/page/product/incident-management/concept/incident-management.html

Question: 16

When using Inbound Email Actions, what happens if an email is received which has no watermark or reference number?

A.New incident created from the message
B.New interaction is created from the message
C.Email is rejected and auto-reply sent to sender
D.New case is created from the message

Answer: A

Explanation:

Here's a detailed justification for why option A, "New incident created from the message," is the most likely outcome when an email with no watermark or reference number is processed by ServiceNow's Inbound Email Actions within the context of IT Service Management (ITSM).

ServiceNow Inbound Email Actions are configured to process incoming emails and perform specific actions based on defined conditions. The presence of a watermark or reference number is a crucial condition often used to identify an email as related to an existing record (like an incident, change request, etc.). If an email lacks these identifiers, ServiceNow's default behavior in a standard ITSM implementation is usually to create a new incident.

This behavior is driven by the need to address potentially new issues or requests raised by users. Without a watermark, the system cannot associate the email with any existing record, so it assumes it's a new request for assistance. This aligns with the core purpose of ITSM: to efficiently manage and resolve IT service requests. The alternative options of creating an interaction or a case are less common defaults for general IT-related emails; incident creation directly addresses potential IT disruptions. Rejecting the email (option C) would be undesirable as it could result in a user's issue going unaddressed.

The specific behavior depends on the configurations of your ServiceNow instance and the existing Inbound Email Actions. ServiceNow administrators can customize these actions extensively, including setting different default behaviors for emails lacking watermarks. However, creating a new incident is the most logical and generally configured default action in the absence of other identifying information within a typical ITSM setup. This ensures that potential IT problems reported via email are logged and addressed.It's crucial to review your specific ServiceNow instance's Inbound Email Actions to confirm the precise behavior.

For more information on ServiceNow Inbound Email Actions, you can refer to the official ServiceNow documentation:

ServiceNow Docs: Inbound Email Actions
ServiceNow Community: Inbound Email Troubleshooting

Question: 17

Under what circumstances, should you use the Communicate workaround Related Link on the Problem record?

A.The workaround is helpful information for the Callers on the Problem's related Incidents (open)

B.The workaround should be published to a knowledge article, visible from the portal

C.The workaround is helpful information for the members of the Problem's Assignment Group

D.The workaround is helpful information for the members of the Problem's Work notes list

Answer: A

Explanation:

The correct answer is A, "The workaround is helpful information for the Callers on the Problem's related Incidents (open)."

Here's the justification: The "Communicate Workaround" Related Link on a Problem record in ServiceNow ITSM is specifically designed to disseminate a temporary fix or procedure to users affected by the underlying problem, particularly those who have open incidents related to that problem. The primary goal is to reduce the immediate impact of the problem on users by providing a temporary solution they can implement themselves or with minimal support intervention.

Option A directly addresses this goal by targeting the callers on the related open incidents. These individuals are actively experiencing the issue, and providing them with a workaround helps them resume their work and reduces the volume of incoming incident tickets.

Option B, publishing to a knowledge article, is a good practice but not the immediate purpose of the "Communicate Workaround" link. Knowledge articles are more for long-term solutions or documenting known issues for broader use. While the workaround could eventually become a knowledge article, the immediate focus is on resolving the problem for current sufferers.

Option C, informing the Problem's Assignment Group, is already assumed as the assignment group is working on the resolution of the underlying problem. The workaround is mainly for immediate impact on users and not targeted to the resolver group.

Option D, informing members of the Problem's Work notes list, is not directly relevant. Work notes are for internal communication among resolving teams, not for distributing information to end-users. The work notes are for internal resolvers to follow the resolution of the problem.

> Therefore, the "Communicate Workaround" functionality is most appropriately used when a temporary fix or workaround is available and would directly benefit end-users who have reported the issue through incidents.

Further reading:

ServiceNow Product Documentation: Look for documentation on Problem Management and Incident Management workflows in ServiceNow's official documentation.

Question: 18

Which interface is designed for tier 1 IT agents who solve internal or external customer issues?

A.ITSM Dashboard
B.IT Service Management Workspace (Agent Workspace)
C.ITIL Homepage
D.Incident Overview

Answer: B

Explanation:

The correct answer is B, IT Service Management Workspace (Agent Workspace). This Workspace is specifically designed for IT agents, particularly those at tier 1, who directly interact with and resolve customer issues related to IT services. Its user interface is optimized for efficiency in incident management, problem management, and service request fulfillment. Key features include contextual information related to the case, real-time collaboration tools, knowledge base integration for quicker resolution, and a streamlined layout that guides agents through the resolution process.

Option A, ITSM Dashboard, typically provides high-level overviews and analytics for management to track performance and identify trends, not direct issue resolution. The ITIL Homepage (Option C) is too generic and doesn't offer the targeted functionality and interface for daily agent tasks. Incident Overview (Option D) is more of a reporting tool and a component potentially found within the Agent Workspace rather than a standalone interface for resolving issues. Agent Workspace (Option B) integrates all these elements into a single, unified view for the agent.

Agent Workspace leverages a consistent user experience across different applications within ServiceNow, improving agent efficiency and reducing training time. The primary goal of Agent Workspace is to reduce the mean time to resolution (MTTR) of incidents and service requests. It achieves this through automation, contextual information, and streamlined workflows tailored for IT service management processes. The multitab navigation and context-aware components like Activity Stream and Timeline Visualization help agents quickly understand the issue history.

For further reading, refer to the official ServiceNow documentation on Agent Workspace:

ServiceNow Agent Workspace: https://www.servicenow.com/products/agent-workspace.html (General overview)
ServiceNow Docs - Agent Workspace: https://docs.servicenow.com/bundle/sandiego-now-platform/page/administer/workspace/concept/workspace-overview.html (Technical Documentation)

Question: 19

When using Agent assist in the Agent workspace, what are examples of possible solutions can be automatically searched and displayed? (Choose five.)

A.Runbook Actions
B.Knowledge
C.SQL Queries
D.Problems
E.Changes
F.Cases
G.Incidents

Answer: BDEFG

Explanation:

The correct answer (BDEFG) identifies the record types that Agent Assist in ServiceNow's Agent Workspace can automatically search and display to help agents resolve issues quickly. Agent Assist leverages contextual information from the active record (e.g., an Incident) to proactively suggest relevant solutions.

B. Knowledge: Knowledge articles are a crucial resource for resolving incidents. Agent Assist searches for articles related to the incident's description, category, and other fields, providing agents with immediate access to known solutions.

D. Problems: Displaying related Problem records can help agents understand if the incident is part of a larger known issue, potentially leading to faster resolution by linking to an existing workaround or permanent fix.

E. Changes: Relevant Change records provide context about recent system changes that might be causing the incident. This helps agents identify potential root causes related to deployments or configurations.

F. Cases: Cases are similar to incidents, but typically involve more complex or long-term customer issues. Finding related cases may reveal previous solutions or ongoing investigations.

G. Incidents: Finding similar previously resolved incidents is a core function of Agent Assist. This allows agents to leverage the collective knowledge of the organization to quickly resolve common issues by reviewing past solutions.

Options A and C are incorrect. Runbook actions are typically executed via automated workflows rather than being displayed as informational resources for agents within the Agent Assist panel, although integration is possible. SQL Queries are not directly surfaced through Agent Assist as they are a backend database function.

Supporting Documentation:

ServiceNow Product Documentation: https://docs.servicenow.com/ (Search for "Agent Assist")

These resources provide in-depth information about the capabilities and configuration of Agent Assist in

ServiceNow. They can be consulted for more details on the record types that can be searched and displayed.

Question: 20

Which module is a useful starting point for a manager to view current state operational information for Incident management?

A.CMDB Health Dashboard

B.Incident > Overview

C.Manager Workspace

D.Critical Incidents Map

Answer: B

Explanation:

The correct answer is **B. Incident > Overview**.

The "Incident > Overview" module within ServiceNow IT Service Management (ITSM) provides a consolidated dashboard presenting key operational information about incident management. This dashboard typically includes metrics such as the number of open incidents, average resolution time, incidents by priority, incidents by assignment group, and trends over time. This allows a manager to quickly grasp the current state of incident management performance without navigating through numerous reports or lists. It offers a starting point for deeper analysis and proactive problem management. The dashboard is designed with pre-configured widgets presenting relevant data summaries, making it a convenient first stop.

Here's why the other options are less suitable:

A. CMDB Health Dashboard: While the CMDB's health is important and related to incident management (as configuration items are often linked to incidents), this dashboard focuses on the health and integrity of the CMDB itself, not directly on the operational metrics of incident management.

C. Manager Workspace: Manager Workspace could potentially contain incident management information, but it's more broadly focused on providing a workspace for managers to handle various IT processes and tasks.

It's not as directly focused on providing an instant overview of Incident Management as the "Incident > Overview" module.

D. Critical Incidents Map: While useful for visually displaying the geographical location of critical incidents, this is not a comprehensive overview of the entire incident management process's operational health. It's focused on a specific subset of incidents (critical ones).

Authoritative Links:

ServiceNow Documentation - Incident Management:https://docs.servicenow.com/bundle/quebec-it-service-management/page/product/incident-management/concept/c_IncidentManagement.html
ServiceNow Community:https://community.servicenow.com/ (Search for "Incident Overview Dashboard")

Question: 21

The Problem table is extended from what table?

A.Task

B.Major Incident

C.Outage

D.Problem Task

E.Incident

Answer: A

Explanation:

The correct answer is A. Task. The Problem table in ServiceNow, like many other tables representing work to be done, inherits its core functionality and attributes from the Task table. The Task table is the base class for workflow activities in ServiceNow. Extending from the Task table provides the Problem table with features common to all tasks, such as assignment groups, assignees, state management, priority, activity logs, and work notes. Major Incident (B), Outage (C), Problem Task (D), and Incident (E) are not the direct parent table of the Problem table. While Major Incident and Incident are related to Problem Management, they are not the table from which Problem directly extends. Problem Task is a related table, created to break down a larger problem into smaller, manageable tasks. Thus, the Problem table directly extends from the Task table to inherit the fundamental task management capabilities provided by the platform. This inheritance model promotes consistency and efficiency in workflow management across different ServiceNow applications. You can see the table hierarchy in the ServiceNow platform itself within the system dictionary for the Problem table.

Authoritative link: ServiceNow Docs - Table Administration (This documentation explains the concept of tables and their relationship, which supports the explanation)

Question: 22

The Problem Manager wants the Problem Coordinators to be able to Re-analyze a Completed Problem. Which module could they use to make this change?

A.Problem > Administration » Problem Properties

B.System UI > UI Action Groups

C.State Management > State Models

D.System UI > Form Actions

E.System UI > UI Actions

Answer: A

Explanation:

The correct answer is **A. Problem > Administration » Problem Properties**. Here's why:

The requirement is to allow Problem Coordinators to re-analyze completed Problems. This indicates a need to modify the system's behavior concerning the Problem Management process and, specifically, the transition of Problem records.

Problem Properties is the configuration area that controls various aspects of the Problem Management module, including state transitions and related functionalities. It allows administrators to define rules governing how Problem records move between states and what actions are permitted at each state. To allow re-analysis of completed problems, a property related to the state transition or actions available for "Closed" or "Completed" problems needs to be modified. This can be done in Problem properties.

System UI > UI Action Groups, **System UI > Form Actions**, and **System UI > UI Actions** are relevant for modifying the user interface and defining actions available on forms, but they don't directly control the Problem Management process's fundamental properties. While a UI Action could be created to trigger a re-analysis, it's best practice to configure properties if the functionality is meant to change the default behavior. The "Problem Properties" area is more appropriately used to modify a process workflow like this.

State Management > State Models is used for defining state flows of the system, but the question is not about the state flow, but what happens on a certain state(completed). The state transition logic itself may not need modification; rather, it's about allowing users with the "Problem Coordinator" role to perform a specific action (re-analyze) on a closed Problem.

Therefore, the most suitable module to configure this functionality is the **Problem > Administration » Problem Properties** section, as it directly affects the core configuration of the Problem Management process and enables administrators to specify which actions are permissible based on the Problem's state. This approach allows the Problem Manager to customize the application's behaviour related to Problem states and user roles within the Problem Management module, allowing the coordinator to reopen or reanalyze completed problem.

Further research:

ServiceNow Product Documentation: Problem Management: https://docs.servicenow.com/bundle/utopia-it-service-management/page/product/problem_management/concept/problem-management.html
ServiceNow Community: Search for "Problem Management Properties" to find discussions and examples of how to configure the Problem Management module.

Question: 23

Your Problem Manager has a structured problem management process, which includes a final review of the solution implemented and of the data regarding incident reduction. When a problem is resolved, after implementing a fix, they want the Post Fix Review task to be automatically created and assigned to the Problem assignee.
What feature would you use to meet this requirement?

A.State Model
B.Workflow Dashboard
C.Action Modeler
D.Task Creator
E.Flow Designer

Answer: E

Explanation:

The correct answer is E, Flow Designer. Here's why:

Flow Designer in ServiceNow is a low-code platform for automating processes across the ServiceNow platform.
It allows users to create automated workflows (flows) based on triggers and actions.

Here's a breakdown of why Flow Designer is the best choice and why the others aren't:

Flow Designer (E): This is the ideal solution because it enables the creation of a flow that is triggered when a Problem record's state changes to "Resolved". The flow can then automatically create a "Post Fix Review" task and assign it to the Problem assignee. The visual, drag-and-drop interface simplifies this automation.

State Model (A): State Models define the possible states a record can have and the transitions between them. While important for Problem Management, they don't directly automate task creation based on a state change. State Models do not provide the ability to trigger actions based on state changes.

Workflow Dashboard (B): Workflow Dashboards provide visibility into existing workflows and their performance. They do not create new automated processes.

Action Modeler (C): Action Modeler allows you to model potential impacts of a change to a service offering or business process. It's unrelated to automating task creation within Problem Management.

Task Creator (D): While the name might sound relevant, there is no feature actually called "Task Creator" in ServiceNow by default. Even if there were, it would likely lack the flexibility and trigger-based automation that Flow Designer offers.

Flow Designer's ability to create automated, triggered actions makes it the perfect tool to satisfy the requirements of the Problem Manager and streamline their post-fix review process. Flow Designer handles the orchestration and assignment of tasks using a structured and automated procedure.

Authoritative Link:

ServiceNow Documentation: https://docs.servicenow.com/bundle/utah-platform-administration/page/administer/flow-designer/concept/flow-designer.html

Question: 24

Your customer needs help defining Category values for the Problem records. What approach should you suggest? (Choose two.)

A.Re-use existing categories from legacy systems

B.Define categories based on the customer's CMDB classes

C.Re-use existing categories from incident management

D.Define categories based on ITIL problem taxonomy

Answer: BC

Explanation:

The correct answer is **BC** because reusing existing Incident Management categories and aligning with CMDB classes provide a structured and efficient approach to defining Problem categories in ServiceNow.

Justification:

B. Define categories based on the customer's CMDB classes: The Configuration Management Database (CMDB) provides a structured representation of IT infrastructure and services. Aligning Problem categories with CMDB classes ensures that problems are categorized in a way that reflects the impacted IT components.

This facilitates accurate problem analysis, root cause identification, and ultimately, effective problem resolution. Leveraging CMDB classifications allows for problems to be directly related to specific configuration items (CIs), enabling a clearer understanding of the problem's scope and impact. This approach provides a consistent and traceable relationship between problems and the infrastructure they affect.

C. Re-use existing categories from incident management: Incident Management and Problem Management are closely related processes. Incidents often lead to the identification of underlying problems. Reusing incident categories for problems allows for consistent categorization and facilitates easier correlation between incidents and related problems. This reduces duplication of effort and helps ensure that problems are categorized in a way that is familiar to IT staff who are already using the incident categories. It also aids in trend analysis by enabling reports to be generated that identify recurring incidents which should be addressed as problems.

Why other options are less suitable:

A. Re-use existing categories from legacy systems: While historical data is valuable, directly reusing categories from legacy systems might not align with the current IT environment or the organization's CMDB. It's preferable to analyze the legacy categories and adapt them to the ServiceNow environment based on the CMDB and Incident categories.

D. Define categories based on ITIL problem taxonomy: While ITIL provides a framework, relying solely on ITIL taxonomy might result in generic categories that are not specific enough for the organization's unique IT landscape and services. ITIL should serve as a guide but not a rigid blueprint, requiring adaptation to the customer's specific environment and requirements.

Supporting Concepts:

ServiceNow CMDB: A core component of ServiceNow, the CMDB stores information about all the configuration items (CIs) in an IT environment.

ITIL: A widely adopted framework for IT service management (ITSM) that provides best practices for various IT processes, including Incident and Problem Management.

Authoritative Links:

ServiceNow CMDB:https://www.servicenow.com/products/cmdb.html
ITIL:https://www.axelos.com/certifications/itil-certifications
ServiceNow Problem Management:https://www.servicenow.com/products/problem-management.html

Question: 25

When a user clicks on the Communicate fix UI action on the Problem form, what happens?

A.Fix is written to the Comments field on any Incident associated with the problem, which is On Hold, Awaiting Problem

B.Fix is written to the Work notes field on any Incident associated with the problem, which is Active

C.Fix is written to the Comments field on any Incident associated with the problem, which is Active

D.Fix is written to a draft Knowledge article

Answer: B

Explanation:

The correct answer is **B. Fix is written to the Work notes field on any Incident associated with the problem, which is Active.**

Here's a detailed justification:

The "Communicate Fix" UI action on the Problem form in ServiceNow is designed to efficiently disseminate information about a problem's resolution to related Incident records. Specifically, its function is to update the Work notes field of associated Active Incidents.

Why Work Notes? Work notes are internal-facing fields intended for communication and collaboration among IT support staff. This makes them the ideal place to record details of the problem's fix, as it allows technicians working on related incidents to understand the solution implemented. Comments field is an external facing field.

Why Active Incidents? The fix is only relevant to incidents that are currently active and unresolved. Incidents in a closed or resolved state do not need to be updated with this information.

UI Action Logic: The "Communicate Fix" UI action typically runs a script that queries for all active incidents related to the Problem record. It then iterates through these incidents and appends the "Fix notes" entered by the user to the Work notes field of each incident.

Business Value: This functionality saves time and effort by allowing problem managers to communicate the fix to all relevant incidents with a single action, rather than manually updating each incident individually.

Therefore, option B accurately describes the behavior of the "Communicate Fix" UI action.

Authoritative Links for Further Research:

ServiceNow Product Documentation: While direct links to specific documentation are subject to change, you can search the official ServiceNow documentation portal (nowlearning.servicenow.com) for "Problem Management," "Communicate Fix," and "Incident Management" for details on these functionalities and their configurations.

ServiceNow Community Forums: The ServiceNow community forums (community.servicenow.com) are also valuable for finding discussions and best practices related to Problem and Incident Management.

Question: 26

Users with which role can Communicate a workaround or fix? (Choose two.)

A.itil_admin
B.problem_coordinator
C.problem_task_analyst
D.problem_admin

Answer: BD

Explanation:

The correct answer identifies roles authorized to communicate workarounds or fixes in ServiceNow IT Service Management (ITSM). Let's examine why options B and D are correct and why A and C are incorrect.

B. problem_coordinator: The problem coordinator is centrally involved in problem management. Their role involves overseeing problem investigation, identifying root causes, and importantly, communicating workarounds and fixes to impacted stakeholders. Communicating workarounds to incident management teams and end-users is a vital part of minimizing the impact of known problems.

D. problem_admin: The problem administrator possesses broad administrative rights over the Problem Management module. This includes defining problem processes, configuring the system, and monitoring overall problem management performance. They have the authority to communicate workarounds or fixes because they are responsible for the effectiveness of the entire process.

A. itil_admin: While the itil_admin role has broad ITIL access, it is generally not directly responsible for the day-to-day communication of workarounds related to specific problems. This role focuses more on configuration and overall system administration, not specific problem resolutions.

C. problem_task_analyst: The problem task analyst focuses primarily on performing tasks within a problem investigation. While they may contribute to identifying workarounds, their role is not usually focused on communicating them broadly to users.

In summary, the problem_coordinator and problem_admin roles are logically aligned with the responsibility of communicating workarounds and fixes due to their oversight and coordination roles within the problem management process.

Reference material: ServiceNow documentation on user roles in ITSM modules. Consult ServiceNow official documentation for the most up-to-date information on roles and permissions.

Question: 27

When a user clicks on the Communicate workaround UI action on the Problem form, what happens?

A.Workaround is written to the Comments field on any open Incident associated with the problem

B.Workaround is written to the Workaround field on any incident associated with the problem

C.Workaround is written to a draft Knowledge article

D.Workaround is written to the Work notes field on any open Incident associated with the problem

Answer: D

Explanation:

The correct answer is **D: Workaround is written to the Work notes field on any open Incident associated with the problem.**

Here's a detailed justification:

The "Communicate workaround" UI action in ServiceNow's Problem Management module aims to disseminate a temporary solution (workaround) for a known issue (Problem) to related active Incidents. The goal is to mitigate the impact of the Problem on users while a permanent fix is being developed.

Option A is incorrect because writing to the "Comments" field is typically reserved for general conversation and updates, not specifically for the formal communication of a workaround. It lacks the specificity needed for workaround communication.

Option B is incorrect because the "Workaround" field on the Incident record is usually populated before the "Communicate workaround" UI action is used. This field is generally populated when someone identifies a solution from the Problem record. The button helps distribute it.

Option C is incorrect because, while workarounds can eventually be incorporated into Knowledge articles, the immediate action of the "Communicate workaround" UI action isn't to create a draft article. The creation of a Knowledge article is a separate, more formalized process typically occurring after a workaround has proven effective and stable.

Option D is correct because the "Work notes" field is the appropriate place for communicating internal-facing information like a workaround to the Incident assignee. The "Work notes" field provides a clear, traceable, and auditable record of actions taken and information shared in the Incident resolution process. By posting the workaround to the "Work notes" field, Incident resolvers are immediately informed of the temporary solution and can implement it to restore service to affected users. Open incidents are targeted because the point of communicating the workaround is to help resolve currently active incidents.

In summary, the ServiceNow platform is designed to facilitate efficient communication between different IT processes. In this specific context, the “Communicate workaround” UI Action is designed to bridge the gap between Problem Management (finding temporary solutions) and Incident Management (resolving user-reported issues) by automatically transmitting the workaround directly to the active Incidents' "Work notes" section, enabling faster and more effective Incident resolution.

Authoritative links:

ServiceNow Documentation - Problem Management: (Search ServiceNow documentation for "Problem Management" within your instance or on the ServiceNow website for the most up-to-date information as documentation varies by ServiceNow release.)
ServiceNow Community Forums: (Search the ServiceNow community forums for discussions on "Communicate Workaround" to see real-world examples and best practices.)

Question: 28

A tester wants to submit a bug report, because they are not able to see the Communicate Fix link under the Related Links on the Problem form.
What do you recommend that they confirm, before submitting the bug report? (Choose two.)

A.Tester is impersonating a user with communications.manager role

B.Tester is impersonating the assignee, which has the problem_coordinator role

C.Tester is impersonating a user with problem_coordinator role

D.The Fix notes field is filled in and saved

Answer: CD

Explanation:

The question focuses on why a tester cannot see the "Communicate Fix" link on a Problem form in ServiceNow ITSM. The "Communicate Fix" link is typically visible only under specific conditions.

Option C, "Tester is impersonating a user with problem_coordinator role," is correct because the problem_coordinator role is often required to manage and communicate fixes related to problems. This role generally has elevated permissions concerning problem management.

Option D, "The Fix notes field is filled in and saved," is also correct. The "Communicate Fix" link usually becomes visible only after the Fix notes field has been populated and the form saved. This is because the communication typically contains the details of the fix, which are captured in this field. The system is designed to avoid sending out communication without fix details.

Option A is incorrect because communications.manager role is not directly relevant to Problem Management and communicate fix functionality.

Option B is also incorrect, because the link to communicate fix does not depend directly on being the "assignee". Impersonating the assignee doesn't guarantee visibility of the "Communicate Fix" link unless they also possess the problem_coordinator role.

Therefore, the tester should verify that they are impersonating a user with the problem_coordinator role and that the Fix notes field is populated before reporting a bug.

Relevant ServiceNow documentation might describe role-based access control concerning Problem Management and the trigger conditions for displaying the "Communicate Fix" link. Investigating the UI action and associated scripts would also reveal the logic controlling the link's visibility.

Authoritative Links (Examples - replace with actual ServiceNow documentation):

ServiceNow Product Documentation (Search for Problem Management Roles and UI Actions):
https://docs.servicenow.com/

Question: 29

Problem and Problem Task records, move automatically from New to Assess states, when which fields are filled? (Choose two.)

A.Short Descriptor

B.State

C.Assigned to

D.Configuration Item

Answer: BC

Explanation:

The correct answer is **BC: State and Assigned to.** Here's a detailed explanation:

Problem and Problem Task records in ServiceNow do not automatically transition from 'New' to 'Assess' simply because the 'Short Description' or 'Configuration Item' fields are populated. The system behavior for state transitions is driven by workflows, business rules, or scripts that react to specific field changes.

The 'State' field's value directly dictates the current stage of the Problem or Problem Task. While it seems self-evident, a change to the State field initiates further workflow actions. The initial state, 'New', typically changes upon criteria being met.

The 'Assigned to' field is more likely to trigger the 'Assess' state transition. When a problem or problem task is assigned to an individual, it signifies that someone is now responsible for actively investigating and evaluating the issue. This assignment logically moves the record from a passive 'New' state to an active 'Assess' state.

Workflows and business rules can be configured such that upon assignment, the state automatically progresses. The act of assigning denotes that assessment is now underway. Therefore, an assignment to the assigned to field can trigger the transition to an assessment state via configured business rules in the ITSM system.

In summary, although not universally guaranteed, the assignment action (filling 'Assigned to') is far more commonly configured to initiate the move to the 'Assess' state than simply filling in a short description or a configuration item. Directly changing the 'State' field also causes a transition but can be considered a separate event from an automatic transition based on filling other fields.

While ServiceNow provides many OOB (Out-of-the-box) capabilities, Problem Management state transitions can be heavily customized. Always review the specific workflow and business rule configurations in your ServiceNow instance.

Relevant ServiceNow Documentation Links:

ServiceNow Problem Management
ServiceNow Workflows
ServiceNow Business Rules

Question: 30

On a Change Approval Definition record, what does the ‘wait for’ condition define?

A.Whether the change approval is sent to an individual user or a group

B.The state the change must be in before the approval notifications can be sent

C.The number or percentage of users from the approval group that must approve the change

D.The fields that must be populated before the approval can be requested

Answer: C

Explanation:

The correct answer is **C. The number or percentage of users from the approval group that must approve the change.**

The "wait for" condition on a Change Approval Definition record in ServiceNow's IT Service Management (ITSM) module specifically determines the criteria for an approval to be considered complete. It dictates how many approvals (either a number or a percentage) from the designated approval group are necessary for the system to proceed with the next stage of the change management process. This functionality provides granular control over the approval process. This controls how the process waits. For example, if the wait for condition is configured to wait for 50% of the group, and the approval group size is 4 members, the change will progress when 2 members have approved it. If it's set to all the change process will move when all members of the group have approved.

Option A is incorrect because it refers to whom the approval is sent, not the criteria for completion. This aspect is typically defined by the 'Approvers' field or related configurations within the approval definition.

Option B is incorrect because the state of the change initiating the approval is generally governed by workflows or business rules triggering the approval process, not the "wait for" condition. The wait for condition is only used after an approval process has started. Option D is incorrect because the fields required before an approval can be requested are typically managed through data policies and UI policies within the change request form, ensuring data integrity before initiating the approval workflow. The "wait for" condition solely focuses on the fulfillment threshold of approvals within a group.

For more details on Change Approval Definitions and related configurations, refer to the official ServiceNow documentation:https://docs.servicenow.com/ (Search for "Change Approval Definitions" within the ServiceNow documentation portal)

Question: 31

In what table are Change records stored?

A.Change [change_task]
B.Change Request [rfc]
C.Change Request [change_request]
D.Change [change]
E.Change [task_change]

Answer: C

Explanation:

The correct answer is C. Change Request [change_request]. This is because in ServiceNow, change management activities, including creating, managing, and tracking changes, are performed within the change_request table. This table extends the Task table, which means change requests inherit properties and functionalities from the Task table.

Option A, Change [change_task], is incorrect. The change_task table stores individual tasks associated with a change request, not the change request itself.

Option B, Change Request [rfc], is incorrect. While "rfc" might colloquially refer to a "Request for Change", the actual table name is change_request, not rfc.

Option D, Change [change], is incorrect. ServiceNow uses a table structure where the specific type of change is generally indicated in the table name. change alone is not a standard table for storing change requests.

Option E, Change [task_change], is incorrect. There is no standard ServiceNow table named task_change that is used for storing change requests.

The change_request table holds key information about the change, such as its type (normal, standard, emergency), priority, risk, impact, state, and approval status. It is central to the Change Management application within ServiceNow. Understanding the correct table to use for change requests is vital for developers and administrators working on change management workflows, reports, and integrations.

Further research:

ServiceNow Docs - Change Management:https://docs.servicenow.com/bundle/sandiego-it-service-management/page/product/change-management/concept/c_ChangeManagement.html
ServiceNow Community:https://community.servicenow.com/

Question: 32

Risk is configured by default, to calculate Risk = High for a change that is scheduled with only 3 days lead time. Your customer’s change policy requires that changes be requested with 5 days lead time.
How would you satisfy this requirement?

A.Update the Risk Property for Insufficient lead time
B.Update the Risk Assessment Matrix for Insufficient lead time
C.Update the Calculate Risk UI Action
D.Update the Risk Matrix for insufficient lead time
E.Update the Risk Condition for Insufficient lead time

Answer: E

Explanation:

The correct answer is **E. Update the Risk Condition for Insufficient lead time.** Here's a detailed justification:

ServiceNow's Risk Assessment module automatically calculates risk based on predefined conditions. When a change request is created with less than the required lead time, the system flags it as high risk. The mechanism driving this is typically a **Risk Condition**. Risk Conditions are configurable rules that trigger risk adjustments based on specific criteria being met.

The key is to modify how ServiceNow defines "Insufficient lead time." Option E directly addresses this. By updating the Risk Condition associated with the "Insufficient lead time" check, you can adjust the lead time threshold (currently 3 days) to the customer's required 5 days. The Risk Condition would contain a script or criteria that evaluates the difference between the change request's planned start date and the request date. If this difference is less than 5 days, the risk is flagged.

Options A, B, C, and D are incorrect:

A. Update the Risk Property: Risk properties are general configurations for the Risk Assessment module, not specific conditions for calculating risk based on data within the change request.
B. Update the Risk Assessment Matrix: The Risk Assessment Matrix typically maps Impact and Urgency to Risk. While important, it doesn't define what causes high risk. It only displays the result of the risk calculation.
Changing the matrix wouldn't change when a change request is classified as high risk due to lead time.
C. Update the Calculate Risk UI Action: UI Actions trigger server-side scripts. While technically feasible to modify the UI Action, it's bad practice and unnecessarily complex. Risk Conditions offer a cleaner, more maintainable, and more configurable approach.
D. Update the Risk Matrix: Risk Matrix typically defines the Risk score based on Impact and Probability, it is not concerned with the calculations to arrive at the Risk score based on a given change request.

Risk conditions, managed within the "Risk Conditions" module in ServiceNow, allow administrators to fine-tune the automatic risk assessment process based on criteria specific to the change request. This makes them the ideal mechanism to modify the lead time threshold.

Supporting Links:

ServiceNow Docs - Risk Conditions:https://docs.servicenow.com/bundle/tokyo-governance-risk-compliance/page/product/grc/concept/risk-assessment-automation.html (This links to a general page about Risk Assessment automation, including information about Risk Conditions)

Question: 33

How are Releases related to Projects?

A.Project tasks and Release tasks are interchangeable

B.Projects can be part of one or more releases

C.Project features are components of a release

D.Projects need to be completed before releases can be defined

E.Projects are used to do root cause analysis for releases

Answer: B

Explanation:

The most accurate answer is **B. Projects can be part of one or more releases.** Here's a detailed justification:

Releases in ServiceNow ITSM represent the deployment of changes to IT services. These changes can stem from various sources, including projects. A large project might necessitate multiple releases to deliver its features incrementally or manage risk. Projects frequently bundle multiple changes into a single, coordinated deployment (release).

Option A is incorrect because project tasks and release tasks serve different purposes and have distinct workflows. Project tasks focus on delivering project deliverables, while release tasks focus on deploying changes into the production environment.

Option C isn't entirely accurate because project features could be a component within a release, but release features are not always project driven. For example, a release could include fixes that were not initiated via a project.

Option D is misleading. While project deliverables might trigger a release, releases can also occur independently of projects for routine maintenance, patches, or urgent fixes. Projects are not prerequisites for releases to be defined or initiated.

Option E is incorrect because root cause analysis is usually associated with incident or problem management, not directly with projects. Root cause analysis may sometimes trigger a project, but projects aren't primarily for root cause analysis.

Therefore, releases are used to deploy the deliverables and features developed during projects, making option B the most accurate choice. One project may have multiple releases and one release may deploy multiple projects.

For further reading, consider exploring these resources:

ServiceNow Official Documentation: Search within the ServiceNow documentation portal for "Release Management" and "Project Management."
ITIL 4 Foundation Publications: Consult the ITIL 4 framework for best practices on Release Management.

Question: 34

What baseline Change Flows support the baseline Normal Change model?

A.Change - Normal - Assess, Change - Normal - Authorize, Change - Normal - Implement Change - Implementation tasks

B.Change - Normal - New, Change - Normal - Review, Change - Normal - Close, Change - Implementation tasks

C.Change - Normal - New, Change - Normal - Assess, Change - Normal - Implement, Change - Implementation tasks

D.Change - Normal - Assess, Change - Normal - Authorize, Change - Normal - Close, Change - Implementation tasks

Answer: A

Explanation:

The correct answer is A, outlining the Change Flows that support the baseline Normal Change model in ServiceNow ITSM: Change - Normal - Assess, Change - Normal - Authorize, Change - Normal - Implement, and Change - Implementation tasks. This reflects the standard progression of a Normal Change within the ServiceNow platform.

A Normal Change typically requires a detailed assessment to understand the potential impact, risk, and resources required. This assessment phase is explicitly supported by the "Change - Normal - Assess" workflow. After the assessment, authorization is crucial. This phase ensures that appropriate stakeholders review and approve the change before it is implemented. The "Change - Normal - Authorize" workflow caters to this authorization process, typically involving CAB (Change Advisory Board) approval. Following
authorization, the implementation phase brings the change into effect. The "Change - Normal - Implement" workflow guides the execution of the change. Finally, the "Change - Implementation tasks" flow manages the individual tasks involved in the implementation, allowing for granular tracking and control.

Options B, C, and D are incorrect as they either include phases that are not sequentially accurate or miss key phases in the standard Normal Change process. For example, "Change - Normal - Review" and "Change -Normal - Close" are related to change management, but do not directly drive the baseline Normal Change model progression as represented by the default workflows. The New workflow is a starting point, but the core is the Assessment, Authorization and Implementation. The Close workflow is part of the change record lifecycle after it goes into production. Therefore, only option A contains the complete set of baseline Change Flows that actively support the Normal Change model from assessment to implementation.

For deeper understanding, refer to the ServiceNow documentation on Change Management: https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/change-management/concept/c_ChangeManagement.html and specifically the details on Normal Changes https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/change-management/concept/change-management-process.html.

Question: 35

Which of the following Change Task Types are available by default? (Choose three.)

A.Planning

B.Testing

C.Review

D.Deployment

E.Verification

Answer: ABC

Explanation:

The correct answer identifying the default Change Task Types in ServiceNow ITSM is **A. Planning, B. Testing, and C. Review.**

ServiceNow's IT Service Management (ITSM) module provides a structured framework for managing IT services, including change management. Change tasks are granular activities performed as part of a larger change request. The default Change Task Types are pre-configured within the platform to support common stages of a change.

Planning: This task type focuses on defining the scope, resources, schedule, and potential impact of the change. It ensures that the change is well-defined before implementation.

Testing: This task type ensures that the change is thoroughly tested in a controlled environment to identify and resolve any issues before deployment.

Review: This task type focuses on evaluating the change request, the testing results and the impact of the change and providing feedback.

While **Deployment** and **Verification** are important aspects of change management, they are typically implemented as distinct steps in the overall change workflow rather than specific, pre-defined Change Task Types. Deployment would often be a phase triggered after a specific change task. Verification would be similar to the overall outcome of the change request. While these two are part of the overall process, the out of box offerings for change task types are Planning, Testing and Review.

In summary, the default Change Task Types—Planning, Testing, and Review—provide a basic structure for organizing and executing tasks within a change request in ServiceNow ITSM.

For more information, you can refer to the ServiceNow documentation on Change Management. These documents often include details about Change Task Types and how they fit into the overall Change Management process. Explore the ServiceNow documentation portal for official guides. Specifically, search for "ServiceNow Change Management" and "Change Task Management."

Question: 36

What is the Business Rule that triggers automatic group assignment on Incident, Problem or Change requests?

A.Populate Assignment Group based on CI/SO

B.Auto-populate ITSM Assignment Groups

C.ITSM Assignment Lookup Rule

D.Automatic Assignment for ITSM

Answer: A

Explanation:

The provided answer, "Populate Assignment Group based on CI/SOB," is the most likely correct answer for automatic group assignment in ServiceNow's ITSM module via Business Rules. Let's dissect why:

Option A makes the most logical sense because a common implementation pattern for ITSM involves automatically assigning incidents, problems, or changes to the appropriate group based on the Configuration Item (CI) and, often, the service offering or business service (SOB) associated with that CI. This is based on the idea that specific teams are responsible for maintaining or supporting particular systems or services. The Business Rule uses information from the CI and SOB fields on the record to determine the correct assignment group.

The logic often embedded within the Business Rule would query a table (custom or OOB) that maps CIs and/or services to assignment groups. When a new Incident (or Problem or Change) is created, this rule runs, looks at the CI and service, finds the corresponding assignment group in the mapping table, and populates the assignment group field.

Option B, "Auto-populate ITSM Assignment Groups," is too vague. While it describes the goal, it doesn't specify how it's achieved. Options C and D are more specific. However, "ITSM Assignment Lookup Rule" is likely a specific type of script include or extension of the functionality described in Option A, not the core Business Rule itself. "Automatic Assignment for ITSM" is similarly too general.

Therefore, the correct and detailed answer is "Populate Assignment Group based on CI/SOB" as the most fitting Business Rule to trigger automatic group assignment.

For further research, consider looking at the following:

ServiceNow documentation on Business Rules:
https://developer.servicenow.com/devportal/$knowledge_base.do#!/article/KB0538057
ServiceNow documentation on CI relationship and automatic assignment: Search ServiceNow Docs for "Assignment Rules based on CI" or "Service-Aware Assignment"

Question: 37

In the CAB workbench, what are some ways the CAB manager can identify the Change requests to be added to a particular meeting agenda? (Choose two.)

A.Change requests meeting different conditions, like Risk level or Type
B.Change requests planned within a certain date range
C.Use any of the options on the Agenda Criteria Tab
D.Change requests for a certain Change Flow Definition

Answer: AB

Explanation:

The provided answer, AB, is the most appropriate choice for identifying Change Requests to add to a CAB meeting agenda within the ServiceNow ITSM CAB Workbench.

Option A is correct because CAB managers often need to prioritize Change Requests based on their potential impact and urgency. Filtering by Risk level (e.g., High, Moderate, Low) allows focusing on Changes that pose the greatest threat to service stability. Similarly, filtering by Type (e.g., Standard, Normal, Emergency) helps categorize Changes for focused discussion, as emergency changes often require more scrutiny.

Option B is correct because Change Requests planned within a specific date range are naturally relevant for discussion in a CAB meeting occurring during or shortly before that period. This ensures timely review and approval of Changes that are about to be implemented, providing an opportunity to identify and address any potential conflicts or dependencies. Prioritizing based on the start or end dates allows the CAB to proactively assess and mitigate risks related to upcoming changes.

Option C, "Use any of the options on the Agenda Criteria Tab," while generally true, is too broad and doesn't specify which capabilities are most essential for Change selection. The criteria tab undoubtedly offers options, but A and B are demonstrably relevant.

Option D is incorrect. Change Flow Definition dictates the lifecycle of a Change, defining the stages a Change Request goes through (e.g., submission, approval, implementation, closure). While knowing the Change Flow is helpful, it isn't a primary criterion for deciding whether a Change Request should be added to a particular CAB meeting agenda. The status of the Change (e.g., in review, scheduled) may be more relevant.

Therefore, filtering based on risk level/type and planned date ranges are direct methods for identifying and prioritizing the most critical Change Requests for CAB review and approval. This ensures the CAB meeting is productive and focused on Changes that require the most attention.

Relevant links for further reading:

ServiceNow CAB Workbench Documentation: (search ServiceNow documentation portal for "CAB Workbench")
ITIL 4 Change Enablement:https://www.axelos.com/ (Search for ITIL 4 Change Management documentation)

Question: 38

A change user complains that with the new Preapproved tab, they have to search through many options to find the Reboot Windows Server change. Since they use this change several times per day, it is inconvenient. What should you suggest to make it easier for the change user?

A.Use the Pin feature

B.Make a Favorite

C.Use the keyword search

D.Drag the change tile to the Navigation pane

Answer: A

Explanation:

The question focuses on optimizing the user experience within the ServiceNow ITSM platform, specifically related to pre-approved change templates. The user wants faster access to a frequently used "Reboot Windows Server" change template within the Preapproved tab.

Option A, using the "Pin" feature, is the most efficient solution. ServiceNow's Pin feature allows users to prioritize and keep frequently used items easily accessible within a section. By pinning the "Reboot Windows Server" change template, it will consistently appear at the top or in a designated area of the Preapproved tab, eliminating the need for scrolling or searching. This direct access saves the user valuable time and effort each time they initiate this specific change.

Option B, creating a "Favorite," while helpful, may not be ideal within the context of the Preapproved tab. Favorites are typically accessed from a separate section in the navigation, requiring an extra click or two compared to the immediate visibility provided by pinning.

Option C, using the keyword search, defeats the purpose of streamlining access. While search is functional, it still requires typing and cognitive effort, which the user is trying to avoid.

Option D, dragging the change tile to the Navigation pane, is not a standard or configurable feature within ServiceNow's Preapproved Change Management.

Therefore, pinning the change template provides the most direct and efficient solution to the user's problem, ensuring quick and easy access to the frequently used "Reboot Windows Server" change. The Pin feature, being readily available within the user interface and requiring minimal configuration, makes it the superior choice. The other options may have their uses in different contexts, but they don't directly address the need for rapid and consistent access to a specific item within the Preapproved tab.

While official ServiceNow documentation explicitly detailing the "Pin" feature within Preapproved Changes may be limited (as feature implementations can change), the general concept of pinning/favoriting in UI/UX design for quicker access to frequently used items is a standard best practice and aligns with ServiceNow's aim for improved usability.

Question: 39

Roles control which users can perform which actions on a change record. What are actions, which cannot be performed by anyone, even an administrator? (Choose two.)

A.Update Change Type on an existing change record
B.Delete a Change record
C.Delete a Standard Change Template
D.Delete CAB Definition

Answer: AC

Explanation:

The correct answer is A and C because certain system-level configurations and historical data manipulations are restricted even to administrators in ServiceNow to maintain data integrity and auditability.

A. Update Change Type on an existing change record: This operation is heavily restricted because changing the Change Type after the record is created can significantly impact the entire change management process. The Change Type dictates the workflow, risk assessment, and approval processes that a change request must follow. Altering it mid-stream would undermine these pre-established processes, potentially leading to inconsistencies, bypassed approvals, and inaccurate reporting. In practical terms, you can't simply switch a "Normal" change to an "Emergency" change after it's been implemented without creating chaos.
ServiceNow's design enforces this to uphold the intended governance and control.

C. Delete a Standard Change Template: Standard Change Templates represent pre-approved, low-risk changes with defined workflows and procedures. These templates are critical for standardization and efficiency in change management. Allowing unrestricted deletion of these templates would create a significant risk of process inconsistencies, errors, and the potential for unauthorized changes to be implemented under the guise of a now-deleted standard change. The system protects these critical configurations to maintain the integrity and consistency of the change management process. While an administrator can deactivate or modify a standard change template, complete deletion is generally prevented to preserve the audit trail and prevent accidental or malicious removal.

Why B and D are not correct:

B. Delete a Change record: While not ideal, administrators, with appropriate roles (e.g., change_manager, change_admin) can delete change records under specific circumstances. The ability might be restricted based on auditing requirements or company policy via script or security constraints.
D. Delete CAB Definition: CAB (Change Advisory Board) Definitions can be deleted, typically by administrators who manage the Change Management application. It's a configuration setting, and deleting it may be a necessary administrative task to adjust the change process.

Therefore, actions related to fundamentally altering the type of a change record or removing pre-defined templates that ensure standardized change processes are protected to ensure best practices, security, and auditing standards within ServiceNow.

Supporting Documentation:

ServiceNow Product Documentation: https://docs.servicenow.com/ (Search for "Change Management Roles", "Standard Change Templates", and "Change Process") This is the direct source of truth for ServiceNow's functionality and configuration. While there is no single page detailing the impossibility of these actions, the design of the platform and the role hierarchy implicitly demonstrate that these actions are not intended to be directly performed through simple administrative privileges. The documentation for change_manager and change_admin role defines the capabilities that these users can do, which is related to closing or cancelling but not deleting or changing change type.

Question: 40

In the baseline Change - Normal model how can Change Tasks be added? (Choose two.)

A.Automatically via the Change - Implementation subflow

B.Manually by the user during New, Assess, and Authorized states

C.Automatically depending on the category selected on the Change Request

D.Manually by the user during all states, except Closed or Canceled

Answer: AD

Explanation:

The correct answer is AD because Change Tasks in the baseline Change - Normal model can be added both automatically and manually, under specific conditions.

Option A is correct because the Change - Implementation subflow, associated with the Change - Normal model, can be configured to automatically create Change Tasks based on pre-defined conditions or activities within the workflow. This automation streamlines the implementation process by ensuring necessary tasks are created without manual intervention. This relates to the cloud computing concept of automation, reducing manual effort and improving efficiency in service management.

Option D is also correct because users can manually add Change Tasks during most states of the Change Request lifecycle, except when the Change Request is in the Closed or Canceled states. This allows for flexibility in managing the change, addressing unexpected issues, or adding tasks that were not initially planned. This embodies the principle of flexibility and control in cloud-based ITSM systems, enabling users to adapt to changing circumstances. Manual creation can happen directly on the Change Request.

Option B is incorrect because while users can manually add Change Tasks during the New, Assess, and Authorized states, their ability to do so is not limited to these states. They can do it in other active states like Implement.

Option C is incorrect as the category of the Change Request, while used for categorization and routing, doesn't automatically trigger the creation of change tasks directly in the baseline model, unless custom workflows are created for this purpose. The baseline workflow primarily uses the Change - Implementation Subflow (A) for automated Change Task generation.

In summary, the baseline Change - Normal model supports both automated and manual addition of Change Tasks, providing a balance between efficiency and control over the change management process.

References:

ServiceNow Documentation: https://docs.servicenow.com/ (Search for "Change Management" and "Change Task")

Question: 41

In the baseline Change - Normal model, when the Change request goes to the Review state, what happens to the implementation and testing tasks, if they have not been closed.

A.They are automatically canceled

B.They are automatically closed

C.They are automatically assigned to the Change assignee and closed

D.An error displays, requiring that the Tasks be closed before moving to Review

Answer: A

Explanation:

The correct answer is A: They are automatically canceled.

Here's why: ServiceNow's Change Management process, specifically within the baseline Change - Normal model, is designed to enforce structured and controlled changes. When a Change request transitions to the Review state, it signifies a shift in focus from active implementation and testing to formal assessment and approval. If implementation and testing tasks remain open at this stage, it indicates a deviation from the planned and expected workflow.

To maintain process integrity and prevent uncontrolled execution of unfinished tasks before proper review, ServiceNow automatically cancels any open implementation and testing tasks associated with the Change request upon entering the Review state. This cancellation prevents these incomplete tasks from being accidentally executed or further modified without approval, which aligns with the overall Change
Management principle of risk mitigation and controlled changes. If these tasks were still needed, they would have to be re-created in the future if the review deemed them important. Options B, C, and D are incorrect because they do not accurately describe the default behavior within the baseline ServiceNow ITSM Change Management process. Option B is incorrect because the tasks are cancelled not closed. Option C is incorrect because the task is not assigned to the Change assignee and closed. Option D is incorrect because the system automatically cancels the open implementation and testing tasks.

Here are some relevant ServiceNow documentation links that support this behavior:

ServiceNow Change Management Documentation (General overview of Change Management)
ServiceNow Change Management Process Flows (Details the process flows, although specific cancellation behavior might require customization)
ServiceNow Task Management (Explains how tasks are managed and how they interact with workflows)

Keep in mind that ServiceNow instances can be highly customized, so the exact behavior might differ based on specific configurations within an organization. However, in a baseline instance, this cancellation of open tasks is the expected outcome when moving a Normal Change request to the Review state.

Question: 42

On the Unauthorized Change Properties module what can you configure? (Choose two.)

A.Enable/Disable creation of Unauthorized changes

B.Maximum number of unauthorized change records for a CI

C.Unauthorized Change Dashboard

D.CI classes to monitor

Answer: AD

Explanation:

The correct answer is AD. The Unauthorized Change Properties module in ServiceNow allows administrators to configure settings related to unauthorized changes in the IT Service Management (ITSM) system.

Option A, "Enable/Disable creation of Unauthorized changes," is a valid configuration option. This setting provides control over whether the system actively identifies and flags changes that haven't followed the proper approval processes. Disabling this feature would prevent unauthorized changes from being detected.

Option D, "CI classes to monitor," is also a valid configuration option. This setting allows administrators to specify which Configuration Item (CI) classes the system should monitor for unauthorized changes. By selecting specific CI classes, such as servers or network devices, the system focuses its monitoring efforts on the most critical assets, preventing an overwhelming number of alerts.

Option B is incorrect because configuring a maximum number of unauthorized change records for a Change Implementation Coordinator (CIC) isn't a direct feature of the Unauthorized Change Properties module.

Option C is incorrect because the properties module doesn't directly configure the Unauthorized Change Dashboard itself. The dashboard configuration involves setting the layout, widgets, and data sources separately.

In summary, the Unauthorized Change Properties module provides settings to enable or disable the creation of unauthorized change records and to define the CI classes that should be monitored for these unauthorized changes. These configurations directly impact how unauthorized changes are detected and managed within the ServiceNow ITSM system.

For further information, review the ServiceNow documentation on Change Management, specifically focusing on unauthorized change settings and properties:

ServiceNow Documentation: (Search "ServiceNow Unauthorized Change Properties" in the ServiceNow documentation portal.)

These resources will provide detailed explanations and examples of how to configure unauthorized change properties in ServiceNow.

Question: 43

How do you describe the relationship between a Knowledge article and a Knowledge base category?

A.Articles can only be published to one category

B.Articles must be published to at least one category

C.Articles must be approved by the selected category owner

D.Articles can be published to a category and subcategory

Answer: A

Explanation:

The correct answer is **A. Articles can only be published to one category.**

Here's a detailed justification:

In ServiceNow's Knowledge Management system, a Knowledge article is primarily associated with a single Knowledge Base category. While articles can be tagged with multiple keywords and may indirectly relate to other areas within your IT service environment, the direct publication and classification of an article are restricted to one specific category. This design promotes a structured approach to knowledge organization, simplifying navigation and search for users seeking solutions within a defined area of expertise.

Option B is incorrect because while it is highly desirable and common practice to publish articles to at least one category to make them accessible, ServiceNow technically allows for articles to exist without a category, although this is not recommended for usability. Option C is incorrect because the approval process is managed through workflows and typically involves knowledge managers or subject matter experts, not necessarily the category owner specifically. Although a category owner might be part of the approval process, it's not a strict requirement. Option D is partially true because a parent category and child categories are hierarchies, and publishing an article in a child category implicitly associates it with the parent, but the explicit publication occurs only in the single selected category.

The structure allows for a clear, hierarchical organization of knowledge, making it easier for users to browse and find the information they need. By limiting the direct association of an article to a single category, ServiceNow maintains a cleaner and more consistent knowledge base structure. This enhances the searchability and maintainability of the knowledge base, contributing to a more efficient IT service management environment. ServiceNow's knowledge management is designed to increase self-service resolution of common problems by having easily findable knowledge. Restricting articles to one primary category helps achieve that.

For further research, you can refer to the official ServiceNow documentation on Knowledge Management:

ServiceNow Docs - Knowledge Management (Replace "quebec" with the appropriate ServiceNow release version if needed).

Question: 44

What are the different ways a user can provide feedback on a knowledge article? (Choose four.)

A.10 Star scale

B.Comment on Article

C.Helpful?

D.Flag Article

E.5 Star scale

F.Pin Article

Answer: BCDE

Explanation:

The correct answer identifies four common mechanisms for users to provide feedback on knowledge articles within ServiceNow's IT Service Management (ITSM) module. Let's break down why each chosen option is correct and why the others aren't:

B. Comment on Article: Allowing users to directly comment on an article enables detailed, qualitative feedback. Users can explain what they found helpful, confusing, or incorrect. This is a crucial feature for iterative improvement of knowledge content.

C. Helpful?: This is a simple binary feedback mechanism (yes/no). It's quick for users and provides a high-level indication of an article's usefulness. Aggregated "helpful" ratings offer a quantitative measure of article effectiveness.

D. Flag Article: Flagging allows users to report issues with an article, such as outdated information, errors, or inappropriate content. This is essential for maintaining the quality and accuracy of the knowledge base. Different flag reasons might also exist, providing further context.

E. 5 Star scale: This provides a more granular quantitative measure of user satisfaction than a simple "helpful" or "not helpful" option. It allows users to rate the article's overall quality and usefulness on a defined scale.

Now, let's examine why the other options are incorrect:

A. 10 Star scale: While theoretically possible to customize, a 10-star scale is not a standard or commonly implemented feature for knowledge article feedback in ServiceNow. A 5-star scale offers sufficient granularity in most scenarios.

F. Pin Article: Pinning is typically a user-specific action that allows them to save an article for easy access later. It's not a mechanism for providing feedback to the knowledge base administrators or authors.

In summary, the chosen options (B, C, D, and E) represent the core functionalities for knowledge article feedback in ServiceNow, enabling both qualitative and quantitative insights into article effectiveness and areas for improvement. These mechanisms directly contribute to the continuous improvement of the knowledge base, ensuring it remains accurate, relevant, and helpful to users.Authoritative links:

ServiceNow Documentation: https://docs.servicenow.com/ (Search for "Knowledge Management" or "Knowledge articles")
ServiceNow Community: https://community.servicenow.com/ (Search for "Knowledge Management feedback")

Question: 45

When using the Knowledge - instant Retire workflow, how does the Valid to date enact a Knowledge article?

A.On Valid to date, article is automatically retired
B.On Valid to date, retire notification is sent to the Knowledge article author
C.On Valid to date, retire notification is sent to the Knowledge base owner
D.On Valid to date, the article is archived

Answer: A

Explanation:

The correct answer is A: "On Valid to date, article is automatically retired."

The "Knowledge - Instant Retire" workflow in ServiceNow is designed for situations where a knowledge article has a defined expiration date. The "Valid to" date field directly controls the article's visibility based on this automated workflow. When the current date matches or exceeds the "Valid to" date, the workflow automatically triggers the retirement process, changing the article's state to "Retired." This automation removes the article from the knowledge base, preventing users from accessing outdated information.

This behavior streamlines knowledge management by ensuring articles are not available beyond their useful lifespan, reducing the risk of users relying on inaccurate or irrelevant content. The alternative options (B and C) involving notifications only, and option D relating to archiving, do not reflect the instant retirement functionality. The intention of the "Knowledge - Instant Retire" workflow is automatic retirement. The "Valid to" date is the trigger and condition for this automated workflow.

More information on Knowledge Management and Workflows can be found within the ServiceNow documentation. Searching for "ServiceNow Knowledge Management Workflows" within the ServiceNow documentation portal should give more insight. Also searching specifically for "Knowledge - Instant Retire" would provide specific information on the workflow itself.

Question: 46

In the ServiceNow native platform, the service catalog can be accessed via the Self-Service > Service Catalog module.
Your customer wants to make modifications to this home page, to add, remove and re-arrange the categories.
Users with what roles can make these edits? (Choose two.)

A.catalog_admin
B.sc_catalog_admin
C.catalog_editor
D.sn_catalog_homepage_write
E.admin

Answer: AE

Explanation:

The correct answer is AE: catalog_admin and admin. Let's justify why.

ServiceNow's access control system dictates who can modify platform features. Making changes to the Service Catalog homepage falls under this system. The catalog_admin role grants broad administrative access specifically related to the Service Catalog. This includes creating, modifying, and managing catalogs, categories, items, and related configurations. Therefore, a user with the catalog_admin role can definitely modify the Service Catalog homepage.

The admin role, often referred to as the "system administrator" role, is the highest privilege in a ServiceNow instance. Users with the admin role bypass most access controls and have the ability to modify almost anything within the instance. Because modifying the Service Catalog homepage is a configuration change, the admin role also provides the necessary permissions.

The sc_catalog_admin role typically focuses on the runtime operations within the service catalog such as reporting and analyzing catalog performance, but not on modifying the categories and the homepage itself.
catalog_editor provides editing abilities within items or categories, but not at the home page level.
sn_catalog_homepage_write is not a valid OOTB (Out-of-the-Box) role related to service catalog homepage modification. Therefore roles B, C and D are incorrect. Only roles A and E possess the rights to make the desired changes to the service catalog home page.

Further research can be done on the ServiceNow documentation site by searching for topics such as "Service Catalog roles," "catalog_admin role," and "admin role." These searches will reveal ServiceNow's official documentation that supports this answer.

Question: 47

What would you use to create a New Hire Employee request which would allow you to order your workstation and company mobile?

A.Knowledge item
B.Record Producer
C.Catalog Item
D.Order Guide
E.Content Item

Answer: D

Explanation:

The correct answer is **D. Order Guide**. Here's why:

An Order Guide in ServiceNow is specifically designed to group multiple catalog items together into a single request. The scenario presented, a New Hire Employee request, inherently involves ordering several related items, such as a workstation and a company mobile. An Order Guide allows you to define rules and dependencies between these items, simplifying the ordering process for the requester and ensuring they receive all necessary items for their role.

Here's a detailed breakdown:

Order Guide Functionality: Order Guides provide a guided experience for users, leading them through the selection of relevant items based on pre-defined rules or conditions. For instance, based on the employee's department, different workstation configurations or mobile phone models could be suggested or required.

Grouping Related Items: The "New Hire Employee" scenario is a perfect example of where an Order Guide excels. It consolidates requests for multiple items, creating a single request containing all necessary components for the new hire. The system automatically adds the catalog items to the user's cart when they submit the order guide, based on rules and the user's selections.

Dependencies and Rules: Order Guides enable the creation of dependencies. For example, ordering a specific type of workstation might automatically include the necessary software licenses. This ensures consistency and completeness in the provisioning process.

Contrast with other options:

Catalog Item: A single item to be requested. While a workstation or mobile phone could each be a Catalog Item, they don't combine multiple items into one user experience.

Record Producer: Used to create records in tables, often to submit incidents or requests. It's not designed to order multiple catalog items.

Knowledge Item: Stores information and articles. Irrelevant to ordering.

Content Item: Used to display content on the Service Portal. Irrelevant to ordering multiple items.

Therefore, an Order Guide's ability to group catalog items, establish dependencies, and guide users through a multi-item request makes it the ideal choice for a New Hire Employee request that includes a workstation and company mobile. The Order Guide will ensure that all required items are requested within the same process.

For further research, consult the official ServiceNow documentation on Order Guides:

ServiceNow Docs - Order Guides

Question: 48

Which tool allows process owners to use natural language to automate approvals, tasks, notifications and other record operations with little to no code?

A.Workflow Mapper

B.Workflow Manager

C.Flow Designer

D.Flow Dashboard

E.Process Designer

Answer: C

Explanation:

The correct answer is **C. Flow Designer**. Here's a detailed justification:

Flow Designer in ServiceNow is specifically designed to enable process owners to automate tasks and processes using natural language constructs, minimizing the need for extensive coding knowledge. It provides a drag-and-drop interface where users can visually build workflows (called Flows) by adding triggers and actions. These actions can include approvals, task creation, notifications, and other record operations within the ServiceNow platform.

The key advantage of Flow Designer is its focus on low-code/no-code development. This lowers the barrier to entry for non-developers and empowers process owners to directly automate their processes without relying heavily on IT specialists for complex scripting. The natural language interface further simplifies the configuration process, allowing users to define actions and conditions in an easily understandable manner. This enhances agility and reduces development time.

Workflow Mapper is related to visualizing existing workflows, not creating new ones with natural language. Workflow Manager isn't a standard ServiceNow tool with this specific functionality. Flow Dashboard provides insights into flow execution, not the design or automation capabilities. Process Designer is focused on high-level process definition and governance, not the detailed automation of tasks within those processes.

Flow Designer provides reusable actions and subflows, promoting consistency and efficiency in automation. It also integrates seamlessly with other ServiceNow components and external systems, expanding its capabilities. By leveraging Flow Designer, organizations can significantly improve process efficiency, reduce manual effort, and enhance the user experience across various IT service management processes.

For further information, refer to the official ServiceNow documentation on Flow Designer:

ServiceNow Documentation - Flow Designer

Question: 49

What process is responsible for defining and managing the lifecycle of all catalog items, by producing and maintaining the services in the catalog and ensuring that a central, accurate, and consistent source of data is provided?

A.Service portfolio management
B.Catalog item management
C.Service mapping
D.Service catalog management

Answer: D

Explanation:

The correct answer is D. Service catalog management is the process specifically responsible for defining, maintaining, and managing the lifecycle of all catalog items. This process focuses on producing and maintaining the services offered in the catalog, ensuring that end-users can easily access and request services they need. A key objective is providing a central, accurate, and consistent source of data about available services. The service catalog acts as a single source of truth for users looking for IT services, including detailed descriptions, pricing, service level agreements (SLAs), and ordering information. It helps streamline the service request process, improves user satisfaction, and ensures that IT services are delivered efficiently.

Service portfolio management (A) deals with a broader view of all services within an organization, including those in the pipeline, active, and retired. It focuses on strategic decision-making regarding investments in services and ensuring alignment with business goals. Catalog Item Management (B) is more focused on managing individual catalog items and their attributes, whereas service catalog management is the broader management of the entire catalog. Service mapping (C) focuses on discovering and documenting the relationships between IT components and the services they support. It's a crucial process for incident and problem management, but it doesn't directly manage the lifecycle of catalog items like service catalog management does. The description in the question clearly emphasizes the management of the entire service catalog and its contents, therefore Service Catalog Management is the correct answer.

Authoritative links:

ServiceNow Documentation on Service Catalog Management: (Search for "ServiceNow Service Catalog Management" on the ServiceNow documentation portal)
ITIL 4 Foundation Publication: (Covers service catalog management as a key practice within IT service management)

Question: 50

Your customer needs different catalogs for:
Human Resources - employee facing - for submitting requests to HR
Customer - external customer facing - for ordering company products and services When these catalogs are created, in which table would the definition be stored?

A.Business Services Catalog [bs_catalog]
B.Catalog [sc_catalog]
C.Service Portfolio Catalog [sc_portfolio]
D.Service Offering Catalog [sn_offering]

Answer: B

Explanation:

The correct answer is B. Catalog [sc_catalog]. Here's a detailed justification:

The ServiceNow table that stores the definition of catalogs is sc_catalog. Catalogs are used to group related items and services, making it easier for users to find what they need. In this scenario, the customer requires two distinct catalogs: one for Human Resources (HR) requests and another for external customer orders. These catalogs are meant to serve different user groups with specific needs.

The sc_catalog table holds the metadata and configuration information for each catalog, including its title, description, and the items or categories it contains. This table defines the structure and behavior of the catalogs presented to the user. It's the central repository for all catalog definitions in ServiceNow.

Option A, Business Services Catalog [bs_catalog], is related to business services, not the general definition of a catalog for requests and products. Option C, Service Portfolio Catalog [sc_portfolio], pertains to the organization of services offered in a business context, focusing on strategic service management. Option D, Service Offering Catalog [sn_offering], represents a detailed description of a specific service and its associated parameters.

Therefore, the table that definitively stores the definition of both the HR and Customer-facing catalogs is sc_catalog. This table allows for the distinct configuration and management of each catalog, catering to the specific requirements of each user base.

Refer to the official ServiceNow documentation for additional details on the sc_catalog table and catalog definitions: https://docs.servicenow.com/ (Search for "sc_catalog table" or "Service Catalog").

Question: 51

When creating a catalog, which field specifies who can edit, update, and delete catalogs, categories, and catalog items?

A.Manager
B.Contributors
C.Owner
D.Editors

Answer: A

Explanation:

The correct answer is A. Manager. Here's a detailed justification:

In ServiceNow's Service Catalog, the "Manager" field on the Catalog record dictates who has the authority to edit, update, and delete the catalog itself, its categories, and the catalog items within it. This role signifies overall responsibility and control over the catalog's content and structure. The "Manager" has broad administrative privileges. While other roles can interact with the catalog, such as adding or using catalog items, their permissions don't extend to the overarching management of the catalog's structure and content like the Manager does.

Option B, "Contributors," typically refers to users who can add content to the catalog, such as new catalog items or knowledge articles, but they usually lack the authority to modify the catalog's core structure or delete existing elements. Contributors are focused on creation and addition, not comprehensive management.

Option C, "Owner," is not a standard field used in ServiceNow Catalog configuration to define who has editing or deletion rights. While an "owner" might conceptually exist for a service, the ServiceNow platform uses the "Manager" field to control these permissions.

Option D, "Editors," isn't a specific standard field directly controlling editing, updating, and deleting privileges for the entire Catalog, Categories and Catalog Items. The "Manager" Role grants such rights.

Therefore, the "Manager" field provides the centralized control required for maintaining the integrity and accuracy of the service catalog. The "Manager" role is critical for ensuring that changes to the service catalog are appropriately authorized and aligned with organizational policies.

For further information, consult the official ServiceNow documentation on Service Catalog Management:

ServiceNow Docs - Service Catalog Management

Question: 52

Which type of catalog item may be found in a Service Catalog?

A.Requested Items
B.Record Producers
C.Categories
D.Execution Plans

Answer: B

Explanation:

Here's a detailed justification for why Record Producers are a type of catalog item found in a Service Catalog, and why the other options are incorrect:

Justification:

A Service Catalog in ServiceNow is a self-service application that allows users to request goods and services. It contains various types of items that users can order.

Record Producers are a specific type of catalog item within the Service Catalog. They are used to create records directly in a table, such as incidents, problems, or change requests. They provide a user-friendly interface to submit forms that create these records, abstracting away the complexity of directly interacting with the backend tables. Record Producers are specifically designed to create records.

Requested Items (REQs) are the result of ordering a Catalog Item. They are records that track the fulfillment process of a request made through the Service Catalog, but they aren't the catalog items themselves. They're the outcome of ordering a catalog item.

Categories are used to organize and group catalog items within the Service Catalog, making it easier for users to find what they're looking for. They aren't the items themselves.

Execution Plans (or workflows) are the behind-the-scenes processes that define how a request is fulfilled. While they are related to the service catalog process, they aren't items that a user directly sees and orders. Execution plans define the flow of tasks.

Therefore, Record Producers are directly available inside a Service Catalog.

Authoritative Links:

ServiceNow Docs: https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/service-catalog-management/concept/c_ServiceCatalogManagement.html
ServiceNow Docs - Record Producers: https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/service-catalog-management/concept/record_producers.html

Question: 53

Which of the following are users able to do when configuring stages in Flow Designer? (Choose two.)

A.Display the stages to the requester
B.Create any number of stages
C.Import a copy of a pre-defined stage set
D.Define the stage set in a subflow

Answer: BC

Explanation:

The correct answer is B and C. When configuring stages in Flow Designer for ServiceNow's Certified Implementation Specialist - IT Service Management exam, users have the flexibility to create any number of stages (Option B). ServiceNow's Flow Designer doesn't impose a hard limit on the number of stages, allowing implementers to model the process accurately. This enables a detailed breakdown of a process workflow, improving visibility and tracking. Furthermore, users can import a copy of a pre-defined stage set (Option C). This feature promotes reusability and consistency across different flows. By importing pre-defined stage sets, implementers can leverage existing best practices and standardized processes, saving time and effort during configuration. This is particularly useful when multiple similar processes require the same or a variation of stage definitions. Option A is incorrect because whether or not stages are visible to the requestor usually depends on other configuration options related to the Service Portal or related interfaces, and stage visibility isn't directly tied to configuring stages within the Flow Designer itself. Option D is misleading; while subflows can be integrated into the main flow, the definition of stage sets doesn't necessarily occur within a subflow. Instead, it is directly done within the flow where the stages need to be configured.

Therefore, the ability to create any number of stages, along with the capacity to import copies of predefined stage sets for reusability, constitute the core functionalities available to users when configuring stages in Flow Designer.

Supporting resources:

ServiceNow Flow Designer Documentation: https://docs.servicenow.com/ (Search for "Flow Designer stages")

Question: 54

When creating a catalog, which field specifies who is able to create, modify, and publish items in the catalog?

A.Editors
B.Item Admins
C.Item Owners
D.Authors

Answer: A

Explanation:

The correct answer is **A. Editors.** Here's a detailed justification:

In ServiceNow's Service Catalog, different roles govern access and permissions regarding catalog and catalog item management. The Editors field on a catalog record dictates which users or groups have the authority to create, modify, and publish catalog items within that specific catalog. This role provides broad control over the catalog's content. Editors can essentially manage the lifecycle of catalog items within their assigned catalog.

Item Admins are not a standard field or role in ServiceNow's Service Catalog configuration. The system uses roles more generally. Similarly, Item Owners is not the specifically defined role that determines who manages the creation, modification, and publishing of the items in the catalog. While item owners might exist, their primary responsibilities wouldn't necessarily encompass comprehensive catalog management.

Authors generally refer to users who can submit content, but they often lack the administrative privileges to modify or publish items directly. In the context of the Service Catalog, authorship might be a contributor role rather than a managerial one.

The Editors field is the most direct and standard way to grant users the necessary permissions to manage the catalog's content, specifically enabling them to create, modify, and publish items. This is a crucial aspect of maintaining a well-managed and up-to-date service catalog.

Further research can be conducted on the official ServiceNow documentation related to Service Catalog roles and permissions:

ServiceNow Docs: Service Catalog
ServiceNow Docs: Catalog Definitions (Examine the "Administrators" and "Managers" configurations within a catalog to understand role-based access).

Question: 55

When defining catalog categories and subcategories, what are some good practices to follow? (Choose two.)

A.Align categories with CMDB classes where possible

B.Keep the number of top-level categories to 8-10

C.Remember that items can only be assigned to one category

D.Do not go to deep with subcategories: go only 1-2 levels deep

Answer: BD

Explanation:

The correct answer is BD. Here's why:

B. **Keep the number of top-level categories to 8-10:** A large number of top-level categories creates a confusing user experience. Users will have difficulty navigating and finding the specific service or item they need. Limiting top-level categories enforces a more organized and manageable catalog structure, promoting discoverability and a positive user experience. A well-defined and concise category structure makes it easier for administrators to maintain and update the catalog as well.

D. **Do not go too deep with subcategories: go only 1-2 levels deep:** Deeply nested subcategories can also make navigation difficult. Users may have to click through multiple layers to find what they are looking for. A shallow hierarchy of categories is easier to navigate and understand. It aligns better with intuitive browsing patterns. Excessive depth creates a cumbersome and time-consuming experience that can frustrate users, ultimately reducing catalog adoption.

A. **Align categories with CMDB classes where possible:** While CMDB classes and catalog items may be related, directly aligning them isn't always practical or user-friendly. CMDB classes are designed for technical representation, whereas catalog categories are meant for a user-centric view of services. Forcing a direct alignment can lead to categories that are technically accurate but hard for users to understand.

C. **Remember that items can only be assigned to one category:** Items can be assigned to multiple categories in ServiceNow. Allowing items to appear in multiple categories enhances discoverability and makes it easier for users to find relevant services regardless of their initial search approach. Restricting items to a single category could limit their visibility and potentially lead to users missing services.

Therefore, options B and D represent best practices for creating a user-friendly and efficient service catalog.

Further Reading:

ServiceNow Documentation: https://docs.servicenow.com/ (Search for "Service Catalog Best Practices") ServiceNow Community Forums: (Search for discussions on Service Catalog design and category structure)

Question: 56

In request fulfillment, approvals can be required before a request can be fulfilled. Your customer is worried about requests getting stuck in the process flow, if the approver is on extended absence from the office.
What can you suggest to alleviate this concern? (Choose two.)

A.The approver can use the Delegate module to assign a person to approve on their behalf, while they are away from the office

B.The approver can set their approval notifications to forward to their personal email address

C.The approval can be defined as a group approval, where any member of the group can approve

D.The approver can set their approval notifications to auto-reply with “approved” in the subject line

Answer: AC

Explanation:

The suggested solutions to prevent request fulfillment bottlenecks due to approver absence are A and C.

Option A, "The approver can use the Delegate module to assign a person to approve on their behalf, while they are away from the office," is a valid solution. ServiceNow's delegation functionality allows users to temporarily assign their responsibilities, including approvals, to another user. This ensures that requests requiring their approval can still be processed without delay. This feature directly addresses the problem of approvers being unavailable.https://docs.servicenow.com/bundle/sandiego-platform-administration/page/administer/delegation/concept/c_Delegation.html

Option C, "The approval can be defined as a group approval, where any member of the group can approve," is another effective strategy. By configuring approvals at the group level, any available member of the assigned group can approve the request. This eliminates single points of failure and ensures continuity in the approval process, even if a specific individual is unavailable. The workload is distributed among multiple individuals, maintaining the flow of requests and preventing delays.https://docs.servicenow.com/bundle/sandiego-platform-administration/page/administer/workflow/concept/c_GroupApproval.html

Option B is incorrect because forwarding notifications to a personal email address doesn't solve the core issue of approval. It only informs the approver, but doesn't enable someone else to formally approve the request within the ServiceNow platform.

Option D is incorrect because setting an auto-reply with "approved" is not a valid or secure approval mechanism within ServiceNow. Approvals require authentication and authorization within the platform, and an email auto-reply would be easily spoofed and create a serious security vulnerability. It bypasses the intended workflow and audit trails.

Question: 57

Released in Quebec, what tool enables you to delegate the creation and maintenance of common and simple use case Catalog Items to business users?

A.Catalog Wizard

B.Catalog Designer

C.Catalog Item Builder

D.Catalog Builder

Answer: D

Explanation:

The correct answer is **D. Catalog Builder**.

Here's a justification:

Catalog Builder, released in the Quebec version of ServiceNow, is specifically designed to empower business users to create and manage simple catalog items without requiring extensive technical expertise. It provides a simplified, intuitive interface that guides users through the process of defining the item's details, questions, and workflow. This delegation reduces the burden on IT and enables faster service delivery by allowing business units to self-manage their common service requests. Catalog Builder achieves this by abstracting away the complexities of the full catalog item creation process, such as advanced scripting or complex workflow configurations. This aligns with a core principle of cloud computing – self-service, allowing users to independently access and manage resources without needing direct intervention from IT specialists. The other options are less accurate. Catalog Wizard is a more general term. Catalog Designer encompasses more complex creation than business users would typically handle. Catalog Item Builder is not a formally recognized term for the specific Quebec release tool. In summary, Catalog Builder is the tool released in Quebec to empower business users in the simple catalog item creation process.

Supporting Links:

ServiceNow Quebec Release Notes: While specific documentation directly highlighting this feature from the original Quebec release might be archived, searching "ServiceNow Quebec release catalog builder" will lead to community discussions and documentation outlining the features added around self-service catalog item creation. Consider filtering searches by date (around the Quebec release timeframe) to get more relevant historical context.

ServiceNow Documentation on Catalog Builder (Later Releases): Although released in Quebec, later versions provide better organized documentation on the tool. While the interface might have evolved, the fundamental purpose remains the same.

ServiceNow Community Forums: Search the ServiceNow community forums for discussions about "Catalog Builder Quebec" to see real-world use cases and experiences.

Question: 58

Request fulfillment relies on three record types, Requests, Requested Items, and Catalog Tasks. The lifecycle status of these records is reflected in a combination of state and stage fields. Which status field is set by the flow?

A.Stage on Requested item

B.Status on Request

C.State on Catalog Task

D.State on Requested Item

Answer: A

Explanation:

The correct answer is A. Stage on Requested Item. Here's why:

Request fulfillment in ServiceNow ITSM utilizes Requests (REQ), Requested Items (RITM), and Catalog Tasks (SCTASK). The lifecycle progresses through these records, each having statuses reflected in state and stage fields. Flows, primarily responsible for automating processes and updating records, directly influence the progress visualization, especially for Requested Items.

Stage (on RITM): The Stage field on the Requested Item provides a high-level overview of where the request is in its overall lifecycle (e.g., Waiting for Approval, Fulfillment, Completed). ServiceNow flows are often configured to update this Stage field as the RITM progresses through different stages, providing users with a clear visual representation of the request's progress.

State (on RITM): While the State field on the RITM also reflects its progress, flows are more commonly used to directly manipulate the Stage field for visualization purposes. The state might be influenced indirectly by the flow but isn't the primary target for stage-based progress.

State (on SCTASK): The State field of Catalog Tasks will be updated by the task worker, it is not the primary source for progress. The stage is updated by the system.

Status (on REQ): The Status field on the Request record is automatically updated based on the status of its associated Requested Items. Flows usually don't directly modify the request status; it's derived from the child RITMs.

In summary, flows actively update the Stage field on Requested Items to visually represent the progress of fulfilling a user's request. The Stage field offers a clear, high-level view of the fulfillment process, and flows are designed to automate updates to this field as the request moves through its lifecycle.

Relevant links for further research:

ServiceNow Docs: https://docs.servicenow.com/ (Search for "Request Fulfillment", "Requested Item", "Catalog Task", "Flow Designer")
ServiceNow Community: https://community.servicenow.com/

Question: 59

Your implementation team has a new Business Analyst. They will be attending their first Service Catalog workshop and will be responsible for capturing notes and decisions from the workshop.
What Now Create assets do you recommend they review, to prepare? (Choose two.)

A.Service Catalog and Request Mgmt - Workshop Preparation Guide

B.Service Catalog and Request Mgmt - Process Guide

C.IT Service Management - Typical Challenges and Remediation

D.ITSM - Business Outcomes and Corresponding KPIs

Answer: AB

Explanation:

The correct answer is AB. Here's why:

A. Service Catalog and Request Mgmt - Workshop Preparation Guide: This guide is specifically designed to prepare individuals for Service Catalog workshops. It likely outlines the objectives of the workshop, the topics to be covered, key stakeholders involved, and the type of information that needs to be gathered. Reviewing this guide allows the new Business Analyst to understand the agenda, anticipate potential discussions, and structure their note-taking accordingly. It's the most direct and relevant resource for pre-workshop preparation, helping them actively participate and effectively capture decisions.

B. Service Catalog and Request Mgmt - Process Guide: Understanding the underlying processes for Service Catalog and Request Management is crucial for a Business Analyst attending such a workshop. This guide will detail the workflows, roles, responsibilities, and key activities involved in creating and managing service catalogs and handling requests. By grasping the process flow, the Business Analyst can better understand the context of the discussions during the workshop and ensure their notes accurately reflect the process nuances and decisions made around them.

Why C and D are not the best choices:

C. IT Service Management - Typical Challenges and Remediation: While understanding challenges is valuable, it's not as immediately important as understanding the workshop agenda and core processes for a new Business Analyst tasked with note-taking. This is a broader ITSM topic.

D. ITSM - Business Outcomes and Corresponding KPIs: Understanding business outcomes and KPIs is useful in the long run, but it's not the priority for preparing for a specific workshop. The BA's immediate goal is to document the discussions and decisions related to the process and how it will be designed, not necessarily the high-level strategic goals yet.

Therefore, the combination of the Workshop Preparation Guide and the Process Guide equips the new Business Analyst with the specific knowledge they need to be an effective participant and note-taker in their first Service Catalog workshop. The preparation guide focuses on how to prepare for a workshop in this context, and the process guide focuses on understanding the business processes that drive the activities of the Service Catalog.

Question: 60

Which role would give you access to the CI Class Manager?

A.ecmdb_admin
B.ecmdb
C.class_manager
D.sn_class_manager

Answer: A

Explanation:

The correct answer is **A. ecmdb_admin**.

The ecmdb_admin role provides comprehensive administrative access to the CMDB (Configuration Management Database), including the CI Class Manager. The CI Class Manager is a crucial tool for managing CI (Configuration Item) classes, their attributes, and relationships within the CMDB. This involves defining CI hierarchies, controlling discovery behavior, and generally maintaining the integrity of the CMDB data model.

The ecmdb_admin role is specifically designed for users who need full control over the CMDB, including the ability to define and modify CI classes. Other roles, like ecmdb, may grant read-only access or limited write access, but lack the full administrative privileges needed for the CI Class Manager. Similarly, the class_manager or sn_class_manager role are not standard ServiceNow roles and will not grant access to the CMDB.

Therefore, to effectively utilize the CI Class Manager for managing and customizing CI classes within ServiceNow's CMDB, the ecmdb_admin role is required. This role encompasses the necessary permissions to perform all administrative tasks associated with CI classes.

For authoritative documentation, refer to ServiceNow's official documentation on roles and CMDB administration:

ServiceNow Docs - CMDB Roles
ServiceNow Docs - CI Class Manager

Question: 61

What module do you use to change the setting for the time between incident Resolution and Closure?

A.ITSM Properties
B.System Settings
C.Incident Settings
D.Incident Properties
E.Resolution Properties

Answer: D

Explanation:

The correct answer is D. Incident Properties.

Incident Properties, accessed through the system properties application, hold a collection of settings that govern the behavior of the Incident Management application. These settings directly influence how incidents are handled, including automated actions, user interface configurations, and specific business rules related to incident workflow. Among the configurable properties are those determining the automatic closure behavior, specifically the duration between when an incident is resolved and when it is automatically closed.

Options A, B, C, and E are not ideal. ITSM Properties is too broad and may contain more than just incident properties. System Settings is also a very general term and lacks specific focus. Incident Settings is not a recognized module. Resolution Properties does not exist as the settings are housed with the broader Incident configuration.

The automatic closure functionality ensures that resolved incidents are eventually closed after a defined period, promoting data hygiene and preventing stale records from remaining open indefinitely. The Incident Properties module allows administrators to adjust this timeframe according to the organization's specific needs and Service Level Agreements (SLAs). Changing this timeframe has implications on reporting (as open incidents will appear for longer if the auto-close timeframe is extended), workflow (longer delays before closure potentially impacting related tasks), and user satisfaction (customers may prefer a longer period if they need to reopen the incident).

Therefore, Incident Properties is the specific module to modify the time between incident resolution and automatic closure.

For further research, see the official ServiceNow documentation on configuring Incident Management properties:https://docs.servicenow.com/bundle/utah-it-service-management/page/product/incident-management/concept/c_IncidentManagementProperties.html

Question: 62

By default, when using Inbound actions, what happens if an email is received which has an Incident watermark?

A.Incident SLA clock is un-paused

B.Incident record is updated, per the action's script

C.Auto-reply sent to sender, recommending they use Portal chat

D.Incident record is re-set to state = attention required

Answer: B

Explanation:

The correct answer is B: Incident record is updated, per the action's script.

Here's why: In ServiceNow, Inbound Actions are mechanisms to process incoming emails and take specific actions based on the email content. When an email with an existing Incident watermark (the unique identifier included in email footers to link to a record) is received, ServiceNow recognizes that the email relates to a pre-existing incident.

The default behavior is not to automatically un-pause an SLA (A), send an auto-reply encouraging portal chat (C), or reset the incident state to "attention required" (D). While these could be configured as part of a custom Inbound Action, they are not the default behavior.

The core functionality triggered by recognizing a watermark is that the system will locate the corresponding incident record. Subsequently, the Inbound Action's script will be executed. This script determines exactly what actions are taken on the incident record. The actions could include adding the email to the activity log, updating fields based on the email content (e.g., adding comments, changing the state), assigning the incident to a different user, or any other custom logic defined within the script. The script will typically extract information from the incoming email (such as the sender, subject, body) and use that information to update the incident record. This is what allows for users to interact with an incident by simply responding to an email.

Therefore, the most accurate default behavior is that the incident record is updated according to the logic programmed within the Inbound Action's script.

For further information on Inbound Actions in ServiceNow, consult the official ServiceNow documentation:

Inbound Email Actions - ServiceNow Docs

Question: 63

When using the Email Client, what is the difference between an Email Template and a Quick Message?

A.Email Templates are like forms that can be sent to the caller for completion; Quick Messages are primarily used by the Chat Bot

B.Email Template is defined and automatically applied when the email form launches; Quick Messages are defined and then can be manually applied by the user

C.Email Templates are included with ITSM; Quick Messages are new with Machine Learning

D.Email templates are defined by users with admin role; Quick Messages are defined by users with quick_message_admin role

Answer: B

Explanation:

The correct answer is B. Here's a detailed justification:

Email Templates and Quick Messages are both features within the ServiceNow platform designed to streamline communication, particularly via the Email Client. However, their application and purpose differ.

An Email Template is a pre-defined email structure that can be automatically applied to an email composition form when certain conditions are met or when a user selects a specific template. This automation saves time by pre-filling fields like subject, body, and recipient based on the context. Think of it as a pre-formatted letterhead for a specific purpose. For instance, a "Password Reset Request" template could automatically populate the email with a standard greeting, instructions, and relevant variables. They are configured for automated usage.

Quick Messages, on the other hand, are snippets of text or pre-composed emails that a user can manually insert into an email they are composing. They are readily available for manual selection, offering a convenient way to include frequently used phrases, standard responses, or common information without having to type them out repeatedly. Quick messages are designed for manual insertion during email composition. They are a library of reusable content.

Therefore, option B accurately reflects this distinction: Email Templates are defined and automatically applied when the email form launches based on pre-configured conditions, whereas Quick Messages are defined and then manually applied by the user at their discretion. Option A is incorrect as Email Templates are not forms for callers. Option C is inaccurate about Quick Messages being solely related to Machine Learning. Option D's role-based definition is misleading; both can be configured by admins.

Question: 64

Your customer wants incidents to close automatically 7 days after the incident is resolved. How do you meet this requirement? (Choose two.)

A.Modify the Incident Lifecycle flow to trigger from the Resolved date instead of the Updated date

B.Update the incident_close UI action script

C.From the Incident Properties application, set Enable auto closure of incidents based on Resolution date to Yes

D.Modify the Incident Lifecycle flow to expire after 7 days

Answer: AC

Explanation:

Here's a breakdown of why options A and C are the correct choices, along with explanations of why option B and D are incorrect:

A. Modify the Incident Lifecycle flow to trigger from the Resolved date instead of the Updated date: The Incident Lifecycle flow (part of the Service Level Management application) is the primary mechanism for automating incident state transitions based on time. By default, this flow might trigger from the "Updated" date, meaning any update resets the timer. Modifying it to trigger from the "Resolved" date ensures the timer starts only after the incident is resolved, which is a crucial part of the requirement. This provides precise control over when the automatic closure process begins.

C. From the Incident Properties application, set Enable auto closure of incidents based on Resolution date to Yes: ServiceNow provides built-in functionality via Incident Properties specifically designed for automatic closure after resolution. Enabling this property is the simplest and most recommended way to accomplish automatic closure based on the resolution date. This leverages ServiceNow's out-of-the-box (OOTB) features, reducing the need for custom scripting.

Here's why the other options are incorrect:

B. Update the incident_close UI action script: While you could technically modify the incident_close UI action, this is a less desirable approach. UI actions are primarily for manual actions taken by users. Modifying this script for an automated process introduces potential complexities and maintainability issues. It's generally better to use the dedicated Incident Properties or workflows for automated tasks.

D. Modify the Incident Lifecycle flow to expire after 7 days: "Expiration" within the Incident Lifecycle flow typically relates to the overall lifespan of the incident rather than a specific duration after resolution. While you can set durations within the flow, directly setting a 7-day expiration might not target the specific "Resolved" date trigger. The key is to ensure the flow is triggered by resolution rather than a general expiry date.

In summary, the combination of modifying the Incident Lifecycle flow to trigger from the "Resolved" date and enabling the "auto closure of incidents based on Resolution date" property offers a robust and supported solution for automatically closing incidents 7 days after they are resolved in ServiceNow. This method utilizes the designed functionalities and configurations provided by ServiceNow to address automation needs without resorting to overly complex or unsupported custom scripting solutions.

Authoritative Links:

ServiceNow Product Documentation:https://docs.servicenow.com/ (Search for "Incident Lifecycle" and "Incident Properties")
ServiceNow Community:https://community.servicenow.com/ (Search for discussions on incident auto-closure)

Question: 65

What tools are available to the assignee to help resolve an Incident? (Choose two.)

A.Known Errors
B.Resolutions from similar incidents
C.CI Class Manager
D.Incident Overview Dashboard
E.Enterprise CMDB Dashboard

Answer: AB

Explanation:

The correct answer is AB. Here's why:

A. **Known Errors:** A "Known Error" record documents the root cause of an issue and a potential workaround, if available. If an incident's symptoms match a known error, the assignee can immediately apply the workaround or resolution documented in the known error, saving significant time and effort. This is a fundamental aspect of proactive problem management, feeding directly into incident resolution efficiency.

B. **Resolutions from similar incidents:** ServiceNow's platform allows for the searching and leveraging of previously resolved incidents. If an assignee encounters an incident similar to one previously resolved, they can review the previous resolution details. This provides a starting point, potentially a full solution, or at least valuable insights into troubleshooting steps. This reusability promotes consistent resolution and knowledge sharing.

C. **CI Class Manager:** While CI Class Manager is valuable for managing Configuration Items (CIs) and their attributes, it doesn't directly provide resolution guidance for an incident. It focuses more on the configuration management database (CMDB) structure and data.

D. **Incident Overview Dashboard:** The Incident Overview Dashboard primarily provides metrics and performance data related to incidents. While helpful for monitoring overall incident trends, it doesn't directly assist an individual assignee in resolving a specific incident. It focuses more on aggregate data and trends.

E. **Enterprise CMDB Dashboard:** While the CMDB provides a wealth of information about IT infrastructure, the "Enterprise CMDB Dashboard" is more about the overall health and accuracy of the CMDB. It doesn't offer targeted, actionable steps for resolving individual incidents. Though related to IT service management, its primary focus is CMDB management, not direct incident resolution.

Therefore, utilizing "Known Errors" and reviewing "Resolutions from similar incidents" directly helps an assignee resolve an incident by providing pre-existing solutions or relevant information for faster and more efficient troubleshooting.

For further research:

ServiceNow Product Documentation: (Requires ServiceNow instance access for full content) Search for "Known Error article" and "Incident Management best practices" within the ServiceNow documentation portal.

ITIL 4 Framework: (Not specific to ServiceNow, but provides the underlying concepts) Axelos ITIL 4 Foundation publication covers incident management and knowledge management principles.

Question: 66

Your customer wants to use the Service Catalog to generate task-based records for end-user inquiries. What Service Catalog capability can you use to generate these records?

A.Execution Plans

B.Content Items

C.Catalog Items

D.Record Producers

Answer: D

Explanation:

The correct answer is D, Record Producers. Here's why:

Record Producers in ServiceNow are specifically designed to create task-based records (like incidents, requests, or problems) from the Service Catalog. They provide a user-friendly interface that allows end-users to submit information through a catalog form. This form data then populates the fields of a new record, streamlining the process of initiating tasks based on user input.

Catalog Items, while part of the Service Catalog, are typically used for requesting goods or services. They don't directly facilitate the creation of task-based records based on inquiries in the way a Record Producer does. While a Catalog Item could indirectly trigger a task through workflows, it's not its primary purpose.

Execution Plans aren't directly related to Service Catalog item submissions. They deal with the orchestration and automation of tasks within a workflow, often related to change management or other complex processes, but they are not the entry point for creating records from user inquiries in the Service Catalog.

Content Items are generally used for providing information or linking to external resources. They are part of the Service Catalog but do not create records based on user input.

In summary, Record Producers provide the ideal functionality for converting end-user inquiries submitted via the Service Catalog into task-based records, making them the correct choice.

For further research:

ServiceNow Documentation on Record Producers:https://docs.servicenow.com/bundle/rome-servicenow-platform/page/product/service-catalog-management/concept/c_RecordProducers.html

Question: 67

Which type of catalog item may be found in a Service Catalog?

A.Requested Items

B.Order guides

C.Categories

D.Execution Plans

Answer: B

Explanation:

The correct answer is **B. Order Guides**. Here's why:

ServiceNow's Service Catalog is a self-service application that provides users with a centralized place to request services, report incidents, and order items. The catalog organizes offerings into logical groupings to help users find what they need.

Order Guides are a specific type of catalog item designed to simplify the process of ordering multiple, related items or services. They present a guided process for users, prompting them for information and then automatically adding the necessary individual items to their cart based on their responses. For example, an order guide for onboarding a new employee might include items like a laptop, software licenses, and access badges, all linked and added together in a cohesive request.

Requested Items (A) are records that track the fulfillment process of a catalog item request. They aren't found in the catalog itself but result from ordering a catalog item.

Categories (C) are used to organize the catalog items to make them easier to find. While vital to the Service Catalog's structure, they aren't items within the catalog, rather groupings of items.

Execution Plans (D) are primarily used in change management to define a series of tasks necessary to implement a change. They don't apply to the Service Catalog.

Therefore, Order Guides are an integral part of a Service Catalog, helping users bundle multiple related services or items into a single, streamlined request.

Here are some authoritative links for further research on ServiceNow Service Catalog and Order Guides:

1. **ServiceNow Documentation on Service Catalog:**https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/service-catalog-management/concept/c_ServiceCatalogManagement.html

2. **ServiceNow Documentation on Order Guides:**https://docs.servicenow.com/bundle/vancouver-it-service-management/page/product/service-catalog-management/concept/c_OrderGuides.html

Question: 68

From which table, is the Incident table extended?

A.Task [task]
B.Task [sn_task]
C.Ticket [ticket]
D.Work [sn_work]

Answer: A

Explanation:

The correct answer is A. Task [task]. Here's a detailed justification:

In ServiceNow, the Incident table is a crucial part of the IT Service Management (ITSM) module and inherits its core functionalities from the Task table. The Task table acts as the parent table for many other tables within ServiceNow, serving as a foundational element for workflow management and record tracking.

The Task table contains common fields and functionalities relevant to various types of work, such as assignment groups, assignees, states, priorities, and work notes. Because Incident, Problem, Change Request, and other similar tables extend from Task, they automatically inherit these essential features. This inheritance allows for consistent management and reporting across different work processes.

Extending from the Task table allows the Incident table to leverage existing workflows, business rules, and UI policies defined at the Task level. This promotes code reuse and simplifies the development process. It also ensures uniformity in how tasks are handled across the platform.

The other options are incorrect because the Incident table does not directly extend from the Ticket, Work, or an undefined table named sn_task. The Task table ([task]) provides the necessary framework and attributes that are fundamental to the Incident table's functionality. Choosing Task as the parent ensures that the

Incident table has the necessary fields and functionalities to manage incidents effectively. Understanding the table hierarchy is critical for building custom applications and modifying existing processes within ServiceNow. The correct hierarchical relationship optimizes performance and maintainability.Authoritative Links:

ServiceNow Documentation on Table Extension:https://developer.servicenow.com/devportal-static.do?topic_name=concept-table-inheritance.html
ServiceNow Product Documentation: (Search within the ServiceNow Product Documentation for "Task table" and "Incident table")