DUMPSBOSS.

Certified Implementation Specialist - IT Service Management

ServiceNow CIS-ITSM

Version Demo

Total Demo Questions: 66

Total Premium Questions: 660

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, ITSM Overview	36
Topic 2, Incident Management	96
Topic 3, Problem Management	71
Topic 4, Change Management	111
Topic 5, Request Management	118
Topic 6, Service Level Management	10
Topic 7, Configuration Management	39
Topic 8, Knowledge Management	46
Topic 9, Mix Questions	133
Total	660

QUESTION NO: 1

What is KCS (Knowledge Centered Services)?

A. A bunch of tables strictly pertaining to CSM case articles that focus on mapping articles to Knowledge management

B. A documented methodology to provide a set of best practices for creating and maintaining knowledge

C. A dashboard with specific visualization of the different knowledge bases and categories

D. An application that helps agents and managers to create cases from Knowledge articles

ANSWER: B

Explanation:

KCS (Knowledge-Centered Service) is a documented, collaborative methodology for capturing, improving, and reusing knowledge as support work is performed. Its practices make knowledge creation and maintenance part of an agent's normal workflow: when a support issue is handled, the agent searches for existing knowledge, reuses it where appropriate, improves it if needed, or creates new knowledge when a gap is identified. This keeps knowledge accurate, relevant, and connected to real customer or employee demand rather than treating articles as a separate, occasional documentation task.

In ServiceNow, Knowledge Management can support this operating model by providing controlled knowledge bases, article lifecycles, feedback, search, and workflows that enable teams to create, review, publish, and continually improve useful content. KCS therefore describes a set of best practices and a way of working, not a particular dashboard, application, or collection of database tables. The Consortium for Service Innovation, which developed KCS, describes it as a methodology for creating and maintaining knowledge as a by-product of solving problems. See the KCS overview from the Consortium for Service Innovation and ServiceNow's Knowledge Management documentation.

QUESTION NO: 2

An administrator has been asked to create a replica of the production organization. The requirement states that existing fields, page layouts, record types, objects, and data contained in the fields and objects need to be available in the replica organization.

How can the administrator meet this requirement?

A. Create a developer sandbox.

B. Create a configuration-only sandbox.

C. Create a metadata sandbox.

D. Create a full sandbox.

ANSWER: D

Explanation:

Create a full sandbox. A full sandbox is designed to replicate the production organization as completely as possible for purposes such as comprehensive testing, integration testing, user acceptance testing, and training. It includes the organization's metadata, which covers configuration elements such as custom and standard object settings, fields, page layouts, record types, automation, and other customizations. Crucially, it also includes a copy of production data and related records, satisfying the requirement that data stored in the organization's fields and objects be available in the replica.

This combination of production metadata and production data makes a full sandbox the appropriate choice when the replica must closely mirror the live environment rather than merely reproduce configuration. Salesforce identifies full sandboxes as containing all production data and metadata, subject to the organization's data-storage limits, and supports sandbox templates to control the data selected for copying where applicable. Administrators should still treat the copied data appropriately, especially if it contains sensitive information, by applying data-masking and access-control practices in the sandbox. See Salesforce's Sandbox Types and Templates documentation and Full Sandbox documentation.

QUESTION NO: 3

Article Versioning can be enabled by setting the property "Enable article versioning feature" to Yes.

A. True

B. False

ANSWER: A

Explanation:

True is correct. Knowledge article versioning is controlled through the Knowledge Management system property labeled *Enable article versioning feature*. Setting this property to *Yes* activates version-management behavior for knowledge articles. With versioning enabled, updates to an article can be managed as distinct versions rather than simply replacing the currently published content without a maintained version history. This supports governance and auditability by allowing knowledge managers and authors to track the evolution of an article and work through the appropriate publishing lifecycle when content requires revision.

Article versioning is especially useful when published knowledge must remain reliable while changes are reviewed, because a new or updated version can be managed without losing the prior article record and its historical context. The setting is an instance-level Knowledge Management configuration property, so an administrator enables the feature through the relevant Knowledge Management properties rather than enabling it separately for each individual article. ServiceNow documents Knowledge Management configuration and article lifecycle capabilities in its Knowledge Management product documentation and the related Knowledge Management properties documentation.

QUESTION NO: 4

Which of the following roles has the ability to create and manage user criteria for service catalogs?

A. catalog_criteria_admin

B. Catalog_admin

C. itil_admin

D. calalog_criteria_manager

E. catalog_manager

ANSWER: A

Explanation:

The **catalog_criteria_admin** role is designed specifically to administer user criteria used by Service Catalog. User criteria define which users can access particular catalog content, such as catalogs, categories, catalog items, record producers, and knowledge content. A criteria administrator can create criteria records, define the conditions or user groups that qualify users, and maintain those criteria as catalog-access requirements change. This separation of duties is useful because it lets an organization delegate audience and entitlement management without necessarily granting broad catalog-maintenance permissions. When a user opens the employee portal or Service Catalog, ServiceNow evaluates the applicable user criteria to determine whether the relevant content should be displayed and available to request. Properly managing these criteria helps ensure that users see services appropriate to their department, location, role, company, or group membership. ServiceNow's Service Catalog documentation describes user criteria as the mechanism for controlling access to catalog content, and its role documentation identifies the specialized administrative access used to manage them. See the ServiceNow User Criteria documentation and the ServiceNow roles reference.

QUESTION NO: 5

Which of these are considered configuration base items? (Choose three.)

A. Servers

B. Computers

C. Applications

D. Accessories

E. Business Service

ANSWER: A B C E

Explanation:

In ServiceNow, configuration items are records managed in the CMDB that represent components needed to deliver and support services. Servers and computers are infrastructure configuration items: they represent managed hardware and can carry attributes, relationships, operational status, and lifecycle information. Applications are also configuration items because the CMDB models software applications and their dependencies on infrastructure and services. A Business Service is likewise a configuration item in ServiceNow's CMDB data model; it represents a service delivered to users or customers and can be related to the applications, servers, and other components that support it.

Consequently, the listed server, computer, application, and business-service records all fit the definition of a configuration item. The question's instruction to choose three is inconsistent with the available choices and current ServiceNow CMDB modeling, since four listed records are valid CI types. ServiceNow describes the CMDB as the system that stores information about configuration items and their relationships, supporting impact analysis and service visibility. See the ServiceNow CMDB documentation and ServiceNow's overview of what a CMDB is.

QUESTION NO: 6

Your customer complains that when their users click on the Configuration Item magnifier from the Incident form, that they are overwhelmed by the volume of CIs to choose from. They want to exclude certain types of CIs from the CI lists on the Incident. Problem and Change forms. What do you recommend to your customer?

A. Add a Show field to the base cmdb table: Check the Show box on those CI records they want to display; make reference qualifier to display only the CIs with show=true

B. Use the Principal CI class checkbox, to identify the CI classes that they want visible on the Incident, Problem, and Change forms

C. Create an Access control to hide the unnecessary CIs from the itil users

D. Make a show/hide UI action to show only the desired CIs to the itil users

ANSWER: B

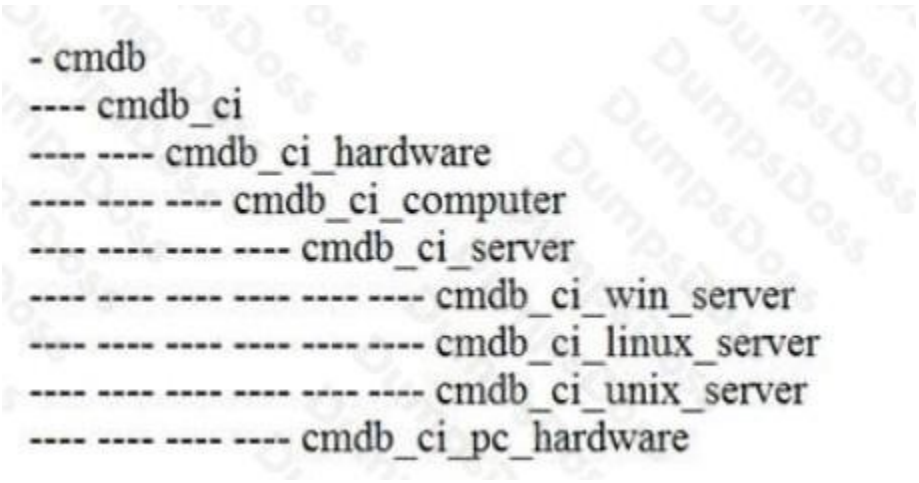
Explanation:

Use the Principal CI class checkbox, to identify the CI classes that they want visible on the Incident, Problem, and Change forms. ServiceNow provides the Principal CI class capability specifically to control which CMDB classes are available through the Configuration item reference field used by ITSM processes. After an administrator marks one or more CI classes as principal, ServiceNow automatically applies the principal-class filter to the CI field on Incident, Problem, and Change records. Consequently, users searching from these forms are presented only with CIs that belong to the designated principal classes, rather than the entire CMDB population.

This is a class-based configuration approach, which directly meets the requirement to exclude particular *types* of CIs while retaining normal reference-field behavior for users. It also keeps the configuration maintainable: administrators can adjust eligibility by updating the relevant CMDB class definition instead of creating custom scripts, UI actions, or record-by-record flags. The setting is available in the CI Class Manager and is intended to identify classes that should be selectable as principal CIs for ITSM records. For implementation details, see ServiceNow's CI Class Manager documentation and its Incident Management documentation.

QUESTION NO: 7

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



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 cmdb_ci_computer and all child classes is correct. In ServiceNow, CMDB classes are implemented as an extended table hierarchy. The `cmdb_ci_computer` table holds records whose class is Computer and can also be extended by more specific CI classes. A report whose source is `cmdb_ci_computer` queries that table and includes records stored in its extension tables, so the report scope includes Computer CIs as well as CIs belonging to every descendant class beneath Computer in the hierarchy.

This behavior lets a report use fields shared by the Computer class while returning a consolidated view across specialized computer CI types. The specific class of each returned record is represented by the `sys_class_name` value, which can be used as a grouping or filtering field when the report needs to distinguish the subclasses. ServiceNow documents that table extension supports a parent table and its child tables, with child tables inheriting the parent's fields. The CMDB data model documentation also describes the CI class hierarchy and how classes extend one another. See the ServiceNow table extension documentation and the CMDB data model documentation.

QUESTION NO: 8

Which of the following options can a survey administrator define on an individual survey? (Choose two.)

A. The ability for end users to decline survey assignments

B. Number of survey reminder notifications

C. Trigger conditions

D. Anonymize responses

ANSWER: B D

Explanation:

Survey administrators can configure **Number of survey reminder notifications** on a survey definition to control the follow-up cadence for users who have received, but not completed, a survey. This setting supports response collection by specifying how many reminder messages may be sent for that particular survey, rather than requiring the same reminder behavior for every survey in the instance. Administrators can therefore apply a more persistent reminder approach to important feedback requests while using a lighter approach for routine surveys.

Anonymize responses is also a survey-level setting. When enabled, it supports confidential feedback by preventing survey results from being associated with an identifiable respondent in the reporting and response data made available to survey administrators. This is especially appropriate for surveys that request candid feedback, such as satisfaction or experience surveys. These settings are part of the survey-definition configuration used to control how a specific survey is delivered and how its submitted data is handled. ServiceNow's survey administration documentation describes the configuration and management of surveys and survey responses: ServiceNow Survey administration documentation. Additional platform documentation is available from the ServiceNow Documentation portal.

QUESTION NO: 9

UC Loans is a small company with a part time Salesforce administrator. UC Loans wants to create a Loan__c record whenever an Opportunity is won.

What is the optimal solution for UC Loans to accomplish this?

A. Process Builder

B. Quick Action

C. Workflow Rule

D. Apex Trigger

E. Record-triggered Flow

ANSWER: E

Explanation:

Record-triggered Flow is the optimal solution because it can run automatically when an Opportunity is updated to the Closed Won stage and create a related `Loan__c` record without custom code. An administrator can configure the flow declaratively by selecting Opportunity as the triggering object, defining entry criteria for the won state, and using a Create Records element to populate the required Loan fields, including a lookup back to the Opportunity. This approach is appropriate for a small organization with limited administrator availability because the automation is centralized, maintainable in Flow Builder, and does not require an Apex developer.

Salesforce identifies Flow Builder as its primary declarative automation tool and recommends using flows for new automation. A record-triggered flow also supports more robust logic and future enhancement if UC Loans later needs approvals, notifications, assignments, or additional data updates as part of the loan-creation process. Configure the flow to run only when the record is updated to meet the won condition so that a Loan record is not created repeatedly during later Opportunity edits. See Salesforce's Flow Builder documentation and its record-triggered flow guidance.

QUESTION NO: 10

Which two characteristics describe a standard change? (Choose two.)

A. It is preauthorized before individual requests are created.

B. It is intended for low-risk and repeatable work.

C. It must be approved by the CAB for every implementation.

D. It is used only when a service outage has already occurred.

ANSWER: A B

Explanation:

A standard change is a low-risk, common, and repeatable change that follows an established procedure. It is preauthorized, which means that it does not require separate approval each time an individual standard change request is submitted.

Organizations still define the implementation steps, risk controls, and review process used to establish a standard change model. See the ServiceNow standard change management documentation.

QUESTION NO: 11

A Portfolio Manager is primarily concerned with the performance of what hierarchy?

A. Portfolio Owner, Service Owner, Catalog Manager

B. Requested Item, Catalog Task, Task SLA

C. Service Catalog, Catalog Item, Requested Item

D. Service Portfolio, Service, Service Offering

ANSWER: D

Explanation:

Service Portfolio, Service, Service Offering is correct because it represents the business-service hierarchy a Portfolio Manager uses to govern and evaluate service performance. The service portfolio is the top-level view of services managed by the organization. Within that portfolio, a service represents a customer-meaningful business or technical capability, while service offerings define the specific variations or commitments through which that service is delivered, such as different support levels, availability commitments, or eligible user populations.

This hierarchy gives the Portfolio Manager a structured way to assess whether services and their offerings remain valuable, aligned to organizational strategy, appropriately funded, and healthy across their lifecycle. It also supports decisions about introducing, changing, investing in, or retiring services based on performance and demand at both the overall service and offering levels. ServiceNow's service portfolio management capability is designed around managing services and service offerings as portfolio entities, providing the appropriate business context for these decisions. See the ServiceNow documentation for Service Portfolio Management and the CSDM product model.

QUESTION NO: 12

Which of the following service catalog variable types are NOT supported in Mobile Classic interfaces? (Choose four.)

A. Label

B. Container Start

C. HTML

D. Lookup Select Box

E. IP Address

F. UI Page

ANSWER: A B C F

Explanation:

Mobile Classic supports a defined subset of Service Catalog variable types so that catalog items can be rendered consistently on mobile devices. **Label**, **Container Start**, **HTML**, and **UI Page** are not supported there because they depend

on presentation constructs or custom browser-oriented rendering rather than a mobile-compatible input control. A Label variable supplies display-only text, while Container Start is used to establish a grouped layout with related container variables. HTML and UI Page variables can introduce custom markup or UI behavior that Mobile Classic does not render as part of its supported catalog-variable experience. When designing a catalog item intended for Mobile Classic, use mobile-supported variable types for data collection and keep instructions or layout requirements within patterns that the mobile client can display reliably. ServiceNow's mobile catalog documentation describes the supported-variable model and the separate constraints that apply to catalog content in the mobile interface. See the ServiceNow Mobile Catalog documentation and the Service Catalog documentation.

QUESTION NO: 13

Even though there are business rules to keep State and Incident State in sync, it is recommended to use Incident State.

A. True

B. False

ANSWER: B

Explanation:

False is correct. For Incident records, the recommended field for implementing and reporting the record lifecycle is *State*, rather than the legacy, incident-specific *Incident State* field. Incident extends the Task table, and *State* is the standard task lifecycle field used consistently by the platform and by other task-based processes. Using this common field supports more consistent configuration, automation, reporting, and integrations across task tables.

Instances can contain both fields for historical compatibility. ServiceNow supplies synchronization logic so that changes made to one can be reflected in the other, helping existing configurations that still rely on *Incident State*. That synchronization does not make *Incident State* the preferred field for new work. When configuring forms, business rules, flows, scripts, filters, and reports, use *State* as the authoritative lifecycle value unless a specific legacy requirement requires otherwise. This approach follows the platform's task-table inheritance model and avoids creating new dependencies on the older incident-only field. See ServiceNow's documentation on table extensions and the Incident Management application.

QUESTION NO: 14

When using the baseline business rule, Populate Assignment Group based on CI/SO, what behavior would you expect on an Incident form? (Choose two.)

A. If selected CI does not have an Owner group, write the Support group from the Service Offering to the Assignment group field

B. If selected CI has a Support group, write that group to the Assignment group field

C. If selected CI has an Owner group, write that group to the Assignment group field

D. If selected CI does not have a Support group, write the Support group from the Service Offering to the Assignment group field

ANSWER: B D

Explanation:

The baseline *Populate Assignment Group based on CI/SO* business rule uses the selected configuration item as the primary source for incident routing. When that configuration item has a value in its Support group field, the rule copies that group into the incident Assignment group field. This provides consistent assignment to the team identified as supporting the affected infrastructure, application, or service component.

The rule also supports a fallback path through the related service offering. If the selected configuration item has no Support group defined, the rule checks the service offering context and populates Assignment group with the Support group specified for that service offering. This allows incidents to be routed even when CI-level support ownership has not been maintained,

while still using service-aware assignment data. The behavior depends specifically on Support group values and the CI/service-offering relationship used by the baseline rule. ServiceNow's ITSM documentation describes using configuration and service data to support incident assignment and fulfillment processes: Incident Management documentation. For CI and service-related records, see CMDB documentation.

QUESTION NO: 15

What is the purpose of the table "Knowledge Use"? (Choose two.)

A. It tracks how many times a knowledge article was viewed.

B. It tracks how many users marked a knowledge article as helpful.

C. It tracks how many pages have links to a knowledge article.

D. It tracks knowledge articles that have been attached to incidents.

E. It tracks how many questions and answers were submitted to a knowledge article.

ANSWER: A D

Explanation:

The Knowledge Use table records meaningful consumption and application of knowledge articles. It supports reporting on article engagement by recording article views, enabling knowledge managers to understand how frequently users access particular content. View information is useful for identifying highly used articles, monitoring adoption of knowledge content, and helping prioritize article maintenance based on actual consumption.

The table also records knowledge use in the context of task work, including when a knowledge article is attached to an incident. This creates a relationship between the article and the task where it was applied, which helps demonstrate that knowledge was used during issue resolution. Those records can be used to analyze which articles support incident handling and to assess the operational value of the knowledge base. ServiceNow Knowledge Management provides reporting and analytics capabilities that use article usage data to help organizations measure knowledge effectiveness and improve content over time. See the ServiceNow Knowledge Management documentation and the Knowledge Management product overview for details on knowledge usage and analytics.

QUESTION NO: 16

Even Though some catalog items approval workflow might be similar, it is recommended to have separate approval workflow for each catalog item.

A. True

B. False

ANSWER: B

Explanation:

False is correct because ServiceNow recommends reusing a catalog-item workflow or flow when catalog items follow the same approval process, rather than creating a separate workflow for every item. A reusable process reduces administrative effort, keeps approval behavior consistent, and makes future changes easier to test and maintain. For example, if several request offerings require the same manager approval followed by fulfillment-task creation, they can use one common process; the process can use catalog-item variables and conditional logic to accommodate appropriate differences. This avoids duplicating substantially identical workflow definitions, which increases the likelihood that one version will be updated while another is missed. Reuse is especially valuable as a catalog grows, because a smaller set of standardized approval processes is easier to govern, troubleshoot, and improve. Separate processes are appropriate only when an item has meaningfully different approval or fulfillment requirements, not merely because it is a different catalog item. ServiceNow's catalog workflow documentation describes associating workflows with catalog items, while its workflow guidance emphasizes

designing maintainable processes and reusing suitable logic. See ServiceNow Catalog Workflows documentation and ServiceNow Workflow administration documentation.

QUESTION NO: 17

Your customer wants to limit the users who ate able to see internal Network requests, to members of the Network department.

Which roles would enable you to make these required changes?

Choose 2 answers

A. catalog_manager

B. catalog_admin

C. user_criteria_admin

D. catalog_editor

ANSWER: B C

Explanation:

`catalog_admin` and `user_criteria_admin` provide the permissions needed to configure Service Catalog visibility for an internal Network request. The intended approach is to create a user criterion that identifies people who belong to the Network department, normally by evaluating group membership, and then associate that criterion with the catalog item's availability settings. The `user_criteria_admin` role permits administration of user criteria, including defining the conditions that determine whether a user is eligible to view or request an item. The `catalog_admin` role provides catalog-administration access needed to maintain catalog items and apply availability controls, such as the available-for user criteria on the Network request item. Together, these roles support both parts of the configuration: defining the eligible audience and enforcing that audience restriction on the relevant catalog content. ServiceNow user criteria are specifically designed to control whether users can see and request catalog items, categories, and related catalog content. See the ServiceNow documentation on user criteria and Service Catalog administration.

QUESTION NO: 18

The Problem table is extended from what table?

A. Outage

B. Problem Task

C. Task

D. Major Incident

E. Incident

ANSWER: C

Explanation:

The **Task** table is correct. In ServiceNow, Problem Management records are stored in the Problem table, whose technical name is `problem`. This table extends the base Task table, `task`. Table extension allows Problem records to inherit the common fields, platform behavior, and task-management capabilities that ServiceNow provides for work items. These inherited capabilities include identifiers such as the task number, assignment information, state handling, activity and audit history, work notes, additional comments, attachments, and relationships to users and groups. The Problem table then adds fields and processes specific to diagnosing the underlying cause of one or more incidents, recording known errors, and managing permanent resolutions. Because it inherits from Task, a Problem record participates in the platform's shared task

framework while retaining its own Problem Management lifecycle and attributes. This parent-child relationship can be verified in the table definition: the Problem table's "Extends table" value is Task. For further information on ServiceNow's table-extension model, see the ServiceNow documentation on extending tables. ServiceNow's Problem Management documentation also describes the application's use of Problem records to investigate and resolve root causes.

QUESTION NO: 19

A Lightning Component functions in preview mode and needs to be used inside a Lightning App Builder page, but it is not available.

What change should be applied to the component?

A. Expose it in the markup using the implements and access attributes.

B. Delete the component, metadata, and Apex controller and recreate them.

C. Refresh the sandbox and upgrade it to the latest API version.

D. Look for errors in the logic in the JavaScript controller.

ANSWER: A

Explanation:

Expose it in the markup using the implements and access attributes. A Lightning Aura component can render correctly when previewed yet still be unavailable in Lightning App Builder because Builder availability is controlled by component metadata declared in the component's `<aura:component>` tag. To make an Aura component selectable for Lightning pages, its markup must implement an appropriate FlexiPage interface, commonly `flexipage:availableForAllPageTypes`. Depending on the intended location, more specific interfaces such as `flexipage:availableForRecordHome`, `flexipage:availableForHomePage`, or `flexipage:availableForAppPage` can be used. The component also needs an access level that permits the relevant Lightning experience to use it; `access="global"` is commonly specified for components intended to be available outside their own namespace. For example, the declaration can include `implements="flexipage:availableForAllPageTypes" access="global"`. After saving the component, it becomes eligible for discovery in the Lightning App Builder component palette, subject to the selected page type and any configured design attributes. Salesforce documents the Aura interface for page availability in its Lightning Component Reference and describes the role of component access in the aura:component reference.

QUESTION NO: 20

What are key relationships between Change and Release Management?

Choose 3 answers

A. A Change can contain one or more Releases

B. A Release can contain one or more Changes

C. Release management application is required, to use the Change management application

D. Change management provides governance which includes Release management

E. Change includes planning and approvals, Release includes building, testing, and execution of changes

ANSWER: B D E

Explanation:

A Release can contain one or more Changes because a release represents a coordinated package of related work that is planned, built, tested, and deployed together. Linking change requests to a release provides a practical way to organize the

individual changes needed to deliver a larger service, application, or infrastructure outcome, while retaining the assessment, authorization, scheduling, and implementation details of each change record.

Change management provides governance which includes Release management because change management establishes the control framework for evaluating risk, obtaining the required approvals, managing collisions and schedules, and maintaining an auditable record of production changes. Release activities operate within that governance so that the deployment of a package is appropriately controlled and aligned with organizational change policy.

Change includes planning and approvals, Release includes building, testing, and execution of changes accurately distinguishes the primary focus of the practices. Change management manages the lifecycle and authorization of changes, while release management coordinates the delivery lifecycle for a group of changes, including preparation, testing, deployment, and validation. ServiceNow documents Change Management and Release Management as related ITSM capabilities that support controlled delivery. See the ServiceNow Change Management documentation and ServiceNow Release Management documentation.

QUESTION NO: 21

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: A D

Explanation:

In the baseline Change - Normal model, change tasks can be generated automatically through the *Change - Implementation* subflow. This Flow Designer subflow supports implementation planning by creating and managing task records associated with the change request, allowing implementation work to be assigned and tracked as part of the controlled change process. Automation through the implementation subflow is particularly useful when task creation must be consistent with an organization's approved implementation workflow.

Change tasks can also be added manually by users while the change request remains active. The baseline model allows users to create and manage related change tasks in all states other than Closed or Canceled. This enables change coordinators, implementers, and assignment groups to add work items as the change progresses through assessment, authorization, scheduling, implementation, and review. Each task remains related to the parent change request, providing visibility into ownership, work progress, and completion before the change is finalized. ServiceNow documents Change Management processes and records at ServiceNow Change Management documentation and describes Flow Designer automation capabilities at ServiceNow Flow Designer documentation.

QUESTION NO: 22

The contextual search feature allows end users to see related knowledge articles, catalog items, resolved incidents and resolved problems to help them solve their problem before they raise an incident.

A. True

B. False

ANSWER: B

Explanation:

False is correct because contextual search is designed to deflect incidents by presenting self-service users with content they can use directly before submitting a record. For end users, the contextual search results available during incident creation

are knowledge articles and catalog items. Knowledge articles can provide instructions, workarounds, and answers to common issues, while catalog items can direct the user to request a service rather than log an incident. This supports a shift-left service experience by helping users resolve straightforward needs through approved self-service resources.

Although contextual search can be configured with different search sources and is also useful in agent workflows, the end-user experience does not expose related incident or problem records as described in the statement. ServiceNow documents contextual search as a capability that searches relevant knowledge and catalog content from the record-production experience, with search sources and visibility governed by the user context and configuration. See the ServiceNow contextual search documentation and the ServiceNow Service Catalog documentation.

QUESTION NO: 23

What is the correct order of steps to follow when working with inbound change sets?

A. Deploy, Validate, Monitor

B. Validate, Deploy, Monitor

C. Deploy, Monitor, Validate

D. Monitor, Deploy, Validate

ANSWER: B

Explanation:

Validate, Deploy, Monitor is the correct sequence for an inbound change set. After an update set is retrieved into the target instance, the administrator first validates it by previewing the inbound update set. Previewing identifies collisions, detects potential issues, and gives the administrator an opportunity to review and resolve conflicts before any records are committed to the target instance. This validation step is essential because update sets can contain changes that overlap with existing target-instance customizations or dependencies.

Once preview results have been reviewed and any necessary collision resolutions or corrective actions are complete, the administrator deploys the changes by committing the update set. Committing applies the captured customer updates to the target instance. The final step is to monitor the deployment and verify that the expected configuration, functionality, and related processes operate correctly after the commit. Monitoring can include reviewing system logs, testing the implemented functionality, and confirming that dependent configuration behaves as intended. ServiceNow documents the inbound update-set process, including previewing and committing, in its Update sets documentation and provides deployment guidance in the Preview an update set documentation.

QUESTION NO: 24

```
Example 1:
AggregateResult[] groupedResults = [SELECT CampaignId, AVG(Amount) FROM Opportunity GROUP BY CampaignId];
for (AggregateResult ar : groupedResults)
{
    System.debug('Campaign ID' + ar.get('CampaignId'));
    System.debug('Average amount' + ar.get('expr0'));
}

Example 2:
AggregateResult[] groupedResults = [SELECT CampaignId, AVG(Amount) theAverage FROM Opportunity GROUP BY CampaignId];
for (AggregateResult ar : groupedResults)
{
    System.debug('Campaign ID' + ar.get('CampaignId'));
    System.debug('Average amount' + ar.get('theAverage'));
}

Example 3:
AggregateResult[] groupedResults = [SELECT CampaignId, AVG(Amount) FROM Opportunity GROUP BY CampaignId];
for (AggregateResult ar : groupedResults)
{
    System.debug('Campaign ID' + ar.get('CampaignId'));
    System.debug('Average amount' ar.get.AVG());
}

Example 4:
AggregateResult[] groupedResults = [SELECT CampaignId, AVG(Amount) theAverage FROM Opportunity GROUP BY CampaignId];
for (AggregateResult ar : groupedResults)
{
    System.debug('Campaign ID' + ar.get('CampaignId'));
    System.debug('Average amount' + ar.theAverage);
}
```

Which two of the examples above have correct System.debug statements? (Choose two.)

A. Example 1

B. Example 2

C. Example 3

D. Example 4

ANSWER: A B

Explanation:

Example 1 and Example 2 contain valid uses of Apex `System.debug`. This method writes a diagnostic message to the Salesforce debug log during execution, making it useful for tracing variable values, confirming code paths, and troubleshooting runtime behavior. Apex supports the basic form `System.debug(message)`, where the message is an expression that evaluates to a value, as well as the form `System.debug(loggingLevel, message)`, which lets developers associate the output with a specific logging level. The valid examples use the method with an allowed argument structure and provide a value that Apex can evaluate and record in the execution log. Debug output is visible only when the applicable user trace flag and log levels are configured, so developers commonly use it alongside the Developer Console, Setup log tools, or API-based log retrieval. Salesforce documents `System.debug` as part of the System class and explains that its output is governed by debug-log settings. See the Salesforce System.debug reference and the Apex debugging documentation for the supported syntax and log behavior.

QUESTION NO: 25

What are the different ways a user can provide feedback on a knowledge article?

Choose 4 answers

A. Helpful?

B. 10 Star scale

C. Comment on Article

D. Pin Article

E. 5 Star scale

F. Flag Article

ANSWER: A C E F

Explanation:

Knowledge Management supports several feedback mechanisms that let readers communicate both the value and the quality of an article to knowledge owners. **Helpful?** records a quick usefulness assessment, helping authors and knowledge managers identify content that meets a reader's need. **Comment on Article** lets a reader supply contextual written feedback, such as a clarification, correction, or improvement suggestion. A **5 Star scale** provides a structured rating that can be aggregated and reviewed as an indicator of article quality and usefulness. **Flag Article** enables readers to bring a potential issue to the attention of the knowledge team, such as inaccurate, outdated, or otherwise problematic content. These feedback inputs support the knowledge lifecycle by giving article owners actionable signals for review and continual improvement. ServiceNow's Knowledge Management documentation describes article feedback and rating capabilities as part of managing article quality and effectiveness; see the ServiceNow Knowledge Management documentation and the ServiceNow product documentation portal.

QUESTION NO: 26

Which of the following fields in the Choice form specifies the field for which this choice is made?

A. Dependent value

B. Element

C. Value

D. Field

ANSWER: B

Explanation:

The correct answer is **Element**. In ServiceNow, a choice is stored as a record in the Choice [sys_choice] table and is associated with a particular field on a particular table. The **Element** field identifies the name of that target field—for example, `impact`, `urgency`, or `state`. This association tells the platform where the defined choice should be available when users interact with records.

Choice records also use the table name to identify the applicable table, while Element narrows that scope to the specific field on the table. Together, these values ensure that a choice such as a display label and its stored value is presented for the intended field. This is particularly important when configuring custom choice lists, extending tables, or managing dependent choices, because the choice must be tied to the correct dictionary element for ServiceNow to render it in the appropriate form control.

ServiceNow documentation describes choices as values available for a field and explains how choice lists are configured through dictionary and choice records. See ServiceNow Choices documentation and ServiceNow Data Dictionary documentation.

QUESTION NO: 27

Category and Subcategory values can be set manually on the Incident form. What are disadvantages of this approach? (Choose two.)

A. Too many options may confuse users and increase mis-categorization

B. Choices have no additional metadata to drive process

C. It is difficult to implement

D. It is not part of the baseline instance

ANSWER: A B

Explanation:

Too many options may confuse users and increase mis-categorization is a disadvantage because manually maintained Category and Subcategory fields generally present users with predefined choice values. As the list grows, agents must interpret similar labels and select the best fit, which can produce inconsistent classifications. Inconsistent classification weakens the quality of incident trend reporting, makes recurring issues harder to identify, and can interfere with routing or assignment logic that relies on a predictable category value.

Choices have no additional metadata to drive process is also a key limitation. A manually selected category or subcategory is primarily a simple choice value rather than a rich record that can carry operational context. Organizations commonly need contextual information such as ownership, service relationships, support groups, lifecycle status, or workflow-relevant attributes to make categorization actionable. When that information is not represented in the selected value itself, it must be maintained separately in custom logic or mappings, increasing administrative effort and making the implementation less scalable. ServiceNow documents choice lists as field-level selectable values and describes how dependency can narrow available selections; this supports the distinction between basic choices and richer service or configuration data used in processes. See ServiceNow Choice Lists documentation and ServiceNow CMDB documentation.

QUESTION NO: 28

Which of the following are valid states for Requested Item Catalog Task? (Choose all that apply.)

A. Closed Skipped

B. Work In Progress

C. New

D. Pending

E. Open

ANSWER: A B D E

Explanation:

Requested Item Catalog Tasks are records in the Catalog Task (sc_task) table that track fulfillment work generated for a requested item. Their out-of-box lifecycle uses task-oriented states, including **Open** when fulfillment work is ready to be addressed, **Work In Progress** while an assignee is actively performing the work, and **Pending** when progress is waiting on an external dependency, information, approval, or other condition. A task can also be set to **Closed Skipped** when the fulfillment activity is intentionally not performed, such as when it is no longer applicable to the request. These values support reporting and workflow logic for catalog fulfillment, letting the platform distinguish unstarted work, active work, temporarily blocked work, and work that was bypassed. ServiceNow documents catalog tasks as fulfillment tasks created from catalog-request processes, and task-state values can be reviewed in the platform's choice-list configuration for the Catalog Task table. See the ServiceNow catalog task documentation and the ServiceNow choice-list administration documentation.

QUESTION NO: 29

A user receives the generic "An internal server error has occurred" while interacting with a custom Lightning Component.

What should the developer do to ensure a more meaningful message?

A. Add an onerror event handler to the tag.

B. Add an error-view component to the markup.

C. Use an AuraHandledException in a try/catch block.

D. Use ProcessBuilder to catch the error.

ANSWER: C

Explanation:

Use an AuraHandledException in a try/catch block. An unhandled exception from an Apex controller method is intentionally surfaced to a Lightning component as a generic internal-server-error message, which avoids exposing implementation details but provides little actionable feedback to the user. The controller should catch an anticipated exception and throw an `AuraHandledException` containing a concise, user-appropriate message. Lightning framework error handling recognizes this exception type and makes its message available to the component's client-side error-handling logic, where the developer can display it in a toast, inline notification, or other approved user interface.

The message should describe the corrective action or relevant business condition without revealing sensitive information such as stack traces, object structure, query details, or system configuration. Developers can also log the original exception server-side before throwing the handled exception, preserving diagnostic detail for administrators while giving the end user a meaningful, safe response. This pattern creates a clear boundary between technical diagnostics and user-facing error communication. Salesforce documents the use of `AuraHandledException` for custom Apex errors in its Apex error-handling guidance and its Lightning Aura custom error documentation.

QUESTION NO: 30

Which of the following roles is allowed to create a new CI class? (Choose two.)

A. itil_admin

B. itil

C. admin

D. cmdb_read

ANSWER: A C

Explanation:

The `itil_admin` and `admin` roles are allowed to create a new CI class. ServiceNow uses the CI Class Manager to manage the CMDB class hierarchy, including creating custom classes and defining their attributes. The `itil_admin` role grants the elevated ITSM administrative capability needed to perform these configuration-management tasks. An administrator can also create CI classes because the `admin` role has broad platform administration privileges, including access to functionality available to elevated administrative roles. Creating a class is a schema and data-model governance activity: it establishes a new type of configuration item, determines its placement in the inheritance hierarchy, and can affect identification, reconciliation, and downstream CMDB processes. Consequently, ServiceNow restricts this action to roles with the appropriate administrative authority. For implementation work, use CI Class Manager to create classes under the appropriate parent class so the new class inherits relevant attributes and behavior consistently. ServiceNow's documentation for CI Class Manager describes managing the CMDB class hierarchy and the required elevated access. Additional CMDB administration guidance is available in the ServiceNow CMDB documentation.

QUESTION NO: 31

Change tasks can manually be added in the following states. (Choose three.)

A. Canceled

B. Implement

C. Authorize

D. Closed

E. Assess

ANSWER: B C E

Explanation:

Change tasks can be manually added while a change request is in the **Assess**, **Authorize**, or **Implement** state. These are active lifecycle stages in which work can still be planned, approved, assigned, and performed. During assessment, teams may create tasks to gather implementation details, assess risk, validate affected services, or complete other preparatory work. During authorization, tasks can support approval-related activities and ensure required work is organized before execution. During implementation, change tasks represent the operational work that fulfiller groups and individual assignees perform to implement the approved change. The change task record provides task-level assignment, state tracking, work notes, and completion information while maintaining its relationship to the parent change request. This enables the overall change state to reflect and govern the work being carried out through its associated tasks. ServiceNow's Change Management documentation describes change tasks as records used to plan and execute work associated with a change request, including manually creating and managing tasks during the applicable active stages of the change lifecycle. See the ServiceNow Change tasks documentation and the Change Management documentation.

QUESTION NO: 32

A company represents their customers as Accounts that have an External ID field called Customer_Number__c. They have a custom Order (Order__c) object, with a Lookup to Account, to represent Orders that are placed in their external order management system (OMS). When an order is fulfilled in the OMS, a REST call to Salesforce should be made that creates an Order record in Salesforce and relates it to the proper Account.

What is the optimal way to implement this?

A. Perform a REST GET on the Account and a REST POST to update the Order__c with the Account's record ID.

B. Perform a REST PATCH to upsert the Order__c and specify the Account's Customer_Number__c in it.

C. Perform a REST POST to update the Order__c and specify the Account's Customer_Number__c in it.

D. Perform a REST GET on the Account and a REST PATCH to upsert the Order__c with the Account's record I

ANSWER: B

Explanation:

Perform a REST PATCH to upsert the Order__c and specify the Account's Customer_Number__c in it. This approach lets the OMS use a stable external business identifier rather than first querying Salesforce to obtain the Account record ID. The Order__c upsert is performed through Salesforce's external-ID sObject resource, allowing an integration to create the order when no matching order exists or update it when one does. In the request payload, the Account lookup can be resolved through the parent relationship by providing the Account external ID—for example, using the relationship name and a nested value such as `Account__r: {"Customer_Number__c": "..."}`. Salesforce resolves that external ID to the matching Account and stores the resulting Account lookup on the order.

This is an efficient, idempotent integration design: retries from the OMS do not need to create duplicate orders when the Order__c upsert key is consistently supplied, and the relationship is established in the same logical write operation. It also avoids coupling the external OMS to Salesforce-generated record IDs, which are not generally meaningful or durable outside Salesforce. Salesforce documents upsert behavior through the external-ID resource in the REST API sObject External ID resource documentation and describes REST sObject operations in the Salesforce REST API developer guide.

QUESTION NO: 33

Which of these roles are allowed to subscribe to CI item? (Choose all that apply.)

A. cmbd_read

B. itil_admin

C. admin

D. itil

ANSWER: B C D

Explanation:

The roles **itil**, **itil_admin**, and **admin** are allowed to subscribe to configuration items. CI subscriptions let qualified IT service-management users follow a CI and receive notifications when relevant activity occurs, helping the teams responsible for a service or component stay informed about changes and related operational events.

The **itil** role provides the baseline ITSM access used by fulfillers who work with configuration items in operational processes. The **itil_admin** role includes ITIL capabilities and therefore also has the access needed to subscribe. The **admin** role has broad platform administrative privileges, including inherited access to ITSM and CMDB functions, so administrators can likewise manage or create CI subscriptions. This reflects ServiceNow's role inheritance model: a role that contains the permissions of another role also receives the capabilities granted through that included role.

For details on ServiceNow roles and role inheritance, see ServiceNow Roles documentation. ServiceNow's CMDB documentation also describes CI-related capabilities and access in the Configuration Management Database documentation.

QUESTION NO: 34

Which table is used to store Q&A data for knowledge articles?

A. kb_qa

B. kb_social

C. kb_social_qa_question

D. kb_social_qa

ANSWER: C

Explanation:

The `kb_social_qa_question` table is the Knowledge Management Social Q&A table used to persist question records associated with knowledge articles. Social Q&A allows users to ask article-specific questions and participate in discussion around published knowledge. A record in this table represents the question and its relationship to the relevant knowledge content; related answer and engagement records support the complete Q&A experience. This structure lets Knowledge Management present questions in the context of an article while retaining the information as managed platform records that can be displayed, searched, and administered.

In ServiceNow, Knowledge Management features are implemented through dedicated application tables rather than by adding unstructured question text directly to the knowledge article record. The Social Q&A data model therefore uses the question table as the central record type for article questions, with associated records handling the related social interactions. This is consistent with ServiceNow's Knowledge Management capabilities and the Social Q&A feature provided for knowledge articles. See the ServiceNow Knowledge Management documentation and the ServiceNow Social Q&A documentation.

QUESTION NO: 35

```
trigger AssignOwnerByRegion on Account ( before insert, before update )
{
    List<Account> accountList = new List<Account>();
    for( Account anAccount : trigger.new )
    {
        Region__c theRegion = [
                SELECT Id, Name, Region_Manager__c
                FROM Region__c
                WHERE Name = :anAccount.Region_Name__c
        ];
        anAccount.OwnerId = theRegion.Region_Manager__c;

        accountList.add( anAccount );
    }
    update accountList;
}
```

Consider the above trigger intended to assign the Account to the manager of the Account's region.

Which two changes should a developer make in this trigger to adhere to best practices? (Choose two.)

A. Use a Map to cache the results of the Region__c query by Id.

B. Move the Region__c query to outside the loop.

C. Remove the last line updating accountList as it is not needed.

D. Use a Map accountMap instead of List accountList.

ANSWER: A B

Explanation:

Use a Map to cache the results of the Region__c query by Id and move the Region__c query to outside the loop. Apex triggers must be bulkified because a single transaction can contain many Account records. Querying Region__c separately while processing each Account consumes one SOQL query per record and can quickly exceed Salesforce governor limits. Instead, the trigger should first collect the relevant Region__c lookup IDs from all incoming Account records, issue one SOQL query using those IDs, and store the returned Region__c records in a map keyed by Id. Each Account can then retrieve its associated region efficiently from that map and assign the Account owner to the region manager. This approach performs a constant number of queries for the transaction rather than scaling queries with the number of records processed. It also provides efficient in-memory lookup of the manager data while iterating through the Accounts. Salesforce's trigger bulkification guidance specifically recommends operating on record collections and avoiding SOQL queries inside loops; its Apex documentation also describes maps as key-based collections suited to retrieving records by ID. See Bulk Trigger Design Patterns and Apex Map Class Reference.

QUESTION NO: 36

Which table can be used to build a report to see which knowledge articles are no longer viewed by users?

A. kb_feedback

B. kb_use

C. kb_article_statistics

D. kb_knowledge

ANSWER: B

Explanation:

The `kb_use` table is the appropriate source for this report because it records knowledge-use activity, including article views. Each time a user accesses a knowledge article, ServiceNow can create a usage record that associates the activity with that article and captures when the use occurred. Reporting on these records makes it possible to group usage by article and evaluate the most recent view date or the number of views during a selected reporting period.

To identify articles that are no longer being viewed, an administrator can use the usage data to find articles with no recent `kb_use` records or with a last-use date older than a defined threshold. This supports knowledge-maintenance activities such as reviewing stale content, validating article relevance, assigning ownership follow-up, or retiring obsolete articles. The table is specifically intended to provide the underlying usage information needed for knowledge analytics, rather than merely storing the article content itself.

ServiceNow documents knowledge-management reporting and analytics capabilities in its Knowledge Management product documentation. Additional platform reporting guidance is available in the ServiceNow Reporting documentation.

QUESTION NO: 37

Change management uses the workflow engine by default to _____. (Choose three.)

A. Send notifications

B. Manage SLAs

C. Generate change tasks

D. Close related tasks

E. Manage approvals

ANSWER: A C E

Explanation:

In Change Management, the workflow engine automates the process steps that move a change through its lifecycle. **Send notifications** is correct because workflow activities can notify assigned users, approvers, implementers, and other stakeholders as a change reaches important states or requires action. **Generate change tasks** is correct because workflows can create and route Change Tasks to the groups or individuals responsible for planning, implementation, testing, and review work. **Manage approvals** is correct because approval activities in a change workflow request, track, and evaluate approvals before the change can proceed to the applicable next stage. These workflow capabilities provide a consistent, auditable process for coordinating the people and work involved in a change. ServiceNow's Change Management documentation describes workflows as the mechanism used to automate change processes, including approvals and task-related activities. The platform workflow documentation also describes workflow activities for approvals, notifications, and record/task creation. See ServiceNow Change Management documentation and ServiceNow Workflow Administration documentation.

QUESTION NO: 38

Technical workshop and process workshop can be combined in one workshop.

A. True

B. False

ANSWER: A

Explanation:

Technical workshop and process workshop can be combined in one workshop. During an ITSM implementation, process decisions and technical design decisions are closely related: a process requirement may affect data design, integrations, security, fulfillment configuration, notifications, or reporting. A combined session enables process owners and technical stakeholders to make those related decisions together, clarify dependencies immediately, and reduce repeated handoffs

between separate meetings. For example, while defining an incident-routing process, the participants can also agree on assignment data, integration requirements, automation considerations, and the technical constraints that affect the proposed workflow.

Combining the workshops does not mean that every workshop must include every stakeholder or cover every implementation topic at once. The workshop should still have a clear agenda, appropriate participants, defined decisions, and documented outcomes. The implementation team can use separate deep-dive sessions where a topic requires focused technical analysis or broader process alignment. ServiceNow's implementation guidance emphasizes structured planning and collaboration with the customer team, while ITSM implementation requires alignment between service-management practices and the platform configuration that supports them. See ServiceNow's implementation success guidance and its overview of IT Service Management.

QUESTION NO: 39

A problem record is the Parent to what record?

A. Workaround

B. Related Incidents

C. Problem Task

D. Known Error

E. Major Incident

ANSWER: C

Explanation:

Problem Task is correct because ServiceNow Problem Management uses problem tasks to organize, assign, and track the work required to investigate, diagnose, remediate, or otherwise progress a problem. A problem record can have one or more associated problem-task records, creating a parent-child relationship: the problem is the parent record and each problem task is a child work item. This structure lets a problem manager divide complex work among different assignment groups or individuals while retaining overall ownership, status visibility, and accountability on the parent problem. For example, separate problem tasks can be created for root-cause analysis, log collection, vendor investigation, testing a permanent fix, and implementing the fix. Updates to those tasks provide operational evidence and progress information that supports management of the overarching problem lifecycle. The relationship is represented through the problem reference on the problem-task record, allowing related tasks to be viewed from the problem and reported on in context. ServiceNow's Problem Management documentation describes the use of problem tasks for carrying out work associated with a problem; see ServiceNow Problem Management documentation and Create a problem task.

QUESTION NO: 40

Universal Containers wants to allow its customers to submit support requests across multiple channels.

In which two ways can this be accomplished with Salesforce? (Choose two.)

A. Expose the case feeds option on the company's website.

B. Offer a Chat button on the company website.

C. Enable Chatter Questions in the company's Customer Community.

D. Enable the Service Cloud Console app on the company website.

ANSWER: B C

Explanation:

Offering a Chat button on the company website provides a real-time digital service channel. Salesforce chat capabilities let website visitors initiate conversations with support agents directly from the site, so customers can obtain assistance without needing to call or email. This supports a channel strategy by connecting customer-facing web interactions to the organization's service operations and agent workspace.

Enabling Chatter Questions in the company's Customer Community provides a community-based support channel. Customers can post questions for answers from other community members, experts, and support personnel. This makes support available in a self-service, many-to-many format and allows useful answers to be retained for future customers who encounter the same issue. Together, website chat and community questions give customers distinct ways to request help: immediate assisted support and collaborative self-service support.

Salesforce describes web messaging and chat as ways to engage customers through digital channels, and its community capabilities support customer participation and knowledge sharing. See Salesforce Trailhead: Service Cloud Communities and Salesforce Trailhead: Service Cloud Chat Basics.

QUESTION NO: 41

Which of the following statements is true when an incident has several child incidents?

A. Updates to State, Comments, Work Notes, and Resolution Code are reflected in child incidents.

B. All updates to parent incident are reflected in child incidents.

C. No updates are synchronized between parent and child incidents.

D. All updates to child incidents are reflected in the parent incident.

ANSWER: A

Explanation:

Updates to State, Comments, Work Notes, and Resolution Code are reflected in child incidents. Parent-child incident relationships are intended to support efficient handling of multiple incidents that arise from the same underlying issue. In this relationship, ServiceNow synchronizes specific parent updates to its child incidents so that affected users and support teams receive consistent status and resolution information without requiring an agent to repeat the same update on every related record.

In particular, changing the parent incident's state communicates the lifecycle progress of the shared issue to the children. Adding comments or work notes to the parent distributes the relevant communication and internal troubleshooting context to those child records. When the parent is resolved, its resolution code is also propagated, maintaining consistent resolution classification across the related incidents. This controlled synchronization is purposeful: it shares the key information needed to manage the common issue while retaining each child incident as an individual record with its own requester and record-specific details. See ServiceNow's Incident Management documentation and the parent-child incident documentation for the relationship behavior and incident-management workflow.

QUESTION NO: 42

Catalog UI Policies may be applied to _____. (Choose two.)

A. Catalog Categories

B. Catalog Items

C. Variable Sets

D. Service Catalogs

ANSWER: B C

Explanation:

Catalog UI Policies may be applied to **Catalog Items** and **Variable Sets**. A Catalog UI Policy controls the runtime behavior of catalog variables presented to a requester, such as whether a variable is visible, mandatory, or read-only. When the policy is associated with a catalog item, its conditions and actions evaluate while that item is being completed in the catalog interface. This enables the item to adapt to information selected or entered by the requester.

Variable Sets are also valid targets because they contain reusable variables that can be included on one or more catalog items. Applying a Catalog UI Policy to a variable set allows the same conditional variable behavior to be defined centrally and reused wherever that variable set is included. For example, a policy can make a variable in a shared variable set mandatory only when a requester chooses a particular service option. This supports consistent requester experiences and reduces duplicated configuration across catalog items. ServiceNow documents Catalog UI Policies as policies that apply to catalog items and variable sets and use Catalog UI Policy Actions to alter variable attributes dynamically. See the ServiceNow Catalog UI Policies documentation and the ServiceNow Variable Sets documentation.

QUESTION NO: 43

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: C

Explanation:

Setting a catalog category to visible cannot be defined through a Catalog UI Policy. Catalog UI Policies are client-side Service Catalog controls that evaluate conditions and then apply actions to variables presented on a catalog item or record producer form. Their purpose is to dynamically control the behavior of those variables during the user's catalog interaction, including whether a variable is mandatory, visible, or disabled (read-only). They do not administer the structure or audience of the service catalog itself.

A catalog category is a navigation and organizational object used to group catalog items. Whether users can access or see a category is governed by the category's configuration and catalog access controls, such as user criteria, rather than by logic acting on item variables after a catalog form loads. This distinction is important in implementations: use Catalog UI Policies for form-level variable experience, and configure category availability through Service Catalog category and user-criteria administration. ServiceNow's documentation describes Catalog UI Policies as a mechanism for setting catalog item variable behavior and documents user criteria as the means to control catalog-content availability. See ServiceNow Catalog UI Policies documentation and ServiceNow User Criteria documentation.

QUESTION NO: 44

Which of the following defines the approvals that should be applied to a change request?

A. Change approval policy

B. Change approval business rules

C. Change approval script includes

D. Change approval UI actions

ANSWER: A

Explanation:

Change approval policy is correct because Change Management uses approval policies to define the approval requirements that apply to a change request. A policy can establish which approvers or approval groups are required, based

on conditions such as the change type, risk, impact, affected service, or other request attributes. When a change satisfies a policy's conditions, ServiceNow evaluates the policy and creates the appropriate approval records as part of the change approval process. This makes approvals consistent, configurable, and aligned with the organization's change-governance requirements.

Approval policies support routing approvals to the right stakeholders for the particular kind of change being assessed. For example, a policy may require additional authorization when a change has a high risk or affects a critical business service, while a lower-risk, preauthorized scenario can follow its defined approval model. Policy conditions and configured approval definitions therefore provide the declarative mechanism that determines the approvals applied to each change request. ServiceNow documents Change approval policies as the configuration used to define and manage approval requirements for changes; see the ServiceNow Change approval policies documentation and the ServiceNow Change Management documentation.

QUESTION NO: 45

Where is the definition of what is provided, or not provided, for a service defined?

A. Vendor service agreements

B. Service scope

C. Service contracts

D. Service limitations

ANSWER: B

Explanation:

Service scope is correct because it establishes the boundaries of a service: the capabilities, activities, deliverables, and support commitments that consumers can expect, along with what is explicitly excluded. Defining this scope gives service owners, support teams, and service consumers a shared understanding of the service's intended coverage. It also provides the baseline for communicating expectations before specific performance targets, availability commitments, or contractual terms are applied. In IT service management, clear service definitions are essential to managing a service portfolio and ensuring that offerings are understandable and consistently delivered. The scope therefore identifies both the included value and the limits of responsibility for the service, which prevents ambiguity about whether a particular request, feature, or outcome is part of the service. ServiceNow's ITSM capabilities support managing services and service offerings as structured portfolio records, while ITIL guidance emphasizes defining services around the value and outcomes they enable for consumers. See ServiceNow IT Service Management and AXELOS: What is ITIL?.

QUESTION NO: 46

How to see which record producers and catalog items are using a certain variable set?

A. In the variable set, check the "Record Producers" related list.

B. Check the Variable Sets Mapping table.

C. None of the listed answers.

D. In the variable set, check the "Included In" related list.

ANSWER: D

Explanation:

In the variable set, check the "Included In" related list. A variable set is a reusable collection of variables that can be attached to multiple Service Catalog items and record producers. The *Included In* related list on the variable set record provides the usage view needed to identify those attachments, so an administrator can quickly determine where a shared variable set is presented and assess the impact of changing its variables, UI policies, or client scripts. This relationship is maintained when

the variable set is included on a catalog item or record producer; therefore, reviewing it from the variable set is the appropriate reverse-lookup method rather than opening each consumer record individually. This is especially useful for impact analysis before modifying a commonly reused set, because the same variable definitions and behaviors can affect every item or producer listed. ServiceNow's Service Catalog documentation describes variable sets as reusable variable collections that are included in catalog items and record producers. See ServiceNow documentation: Variable sets and ServiceNow documentation: Add a variable set to a catalog item.

QUESTION NO: 47

Which of the following can leverage user criteria for controling access?

Choose 2 answers

A. catalog variables

B. catalog taxonomy

C. calalog topics

D. catalog items

E. catalog categories

ANSWER: D E

Explanation:

Catalog items and catalog categories can both use user criteria to control what a requester can access in the Service Catalog. User criteria are reusable conditions that determine whether a user qualifies based on attributes such as role, group membership, company, department, location, or other scripted conditions. Applying an available-for or not-available-for user criterion to a catalog item controls whether that specific requestable offering is displayed and can be ordered by the user. This supports targeted delivery of services, such as showing software requests only to eligible employees or exposing administrative requests only to members of an approved group.

User criteria can also be associated with catalog categories. At the category level, they control visibility of the category and thereby help organize the catalog experience for distinct user populations. Combining category-level visibility with item-level criteria enables an implementation to present a relevant, concise catalog while retaining more granular control for individual offerings. ServiceNow documents user criteria as the mechanism for defining user access to catalog content, including categories and items. See the ServiceNow user criteria documentation and the Service Catalog documentation.

QUESTION NO: 48

Which module would you use to configure Inbound Email Action?

A. System Policy > Inbound Email Actions

B. System Policy > Email > Inbound Actions

C. System Properties > Email Actions

D. System Properties > Email > Inbound Actions

ANSWER: B

Explanation:

System Policy > Email > Inbound Actions is the module used to create, view, and maintain inbound email actions. An inbound email action defines how ServiceNow processes an incoming message after the instance receives it and determines its email type, such as a new message, reply, or forward. Administrators use these records to set conditions that identify applicable emails, specify the target table, and provide the script or field actions that create or update records. For example,

an inbound action can create an incident from a new email or update an existing incident when a customer replies to a notification.

The module belongs under System Policy because inbound actions govern automated platform behavior: they apply processing rules to email received by the instance. Email-related administration is grouped beneath the Email application menu, making the navigation path System Policy > Email > Inbound Actions. ServiceNow's product documentation describes inbound email actions as the mechanism for taking action on received email and documents their configuration, conditions, and scripts. See the ServiceNow inbound email actions documentation and the procedure for creating an inbound email action.

QUESTION NO: 49

Which of the following can be achieved by a user with itil role? (Choose three.)

A. Reconcile CIs

B. Import CIs

C. Create CI

D. Delete CI

E. Subscribe to CI

ANSWER: C D E

Explanation:

The `itil` role provides an IT service-management user with operational access to Configuration Management Database records, including the ability to maintain configuration items directly. Therefore, **Create CI** is correct: ITIL users can add CI records when documenting services, infrastructure, applications, and their related components. **Delete CI** is also correct because the baseline access model permits an ITIL user to remove CI records when they are obsolete, entered in error, or otherwise no longer need to be maintained, subject to any additional organization-specific access-control rules. **Subscribe to CI** is correct because an ITIL user can subscribe to a CI to receive relevant notifications and remain informed about activity or changes involving that configuration item. These capabilities support the ITIL user's responsibility for maintaining accurate operational service data and monitoring the items that affect services they support. In a live instance, administrators can further restrict or extend these capabilities through ACLs, role inheritance, and customizations; however, the question concerns the standard role behavior. See ServiceNow's CMDB documentation and its guidance on roles and access control.

QUESTION NO: 50

Which of the following users has the permission to close an incident? (Choose two.)

A. None of the listed answers

B. Itil Admin

C. Itil User

D. Incident Manager

E. Incident caller

ANSWER: B E

Explanation:

In the standard Incident Management process, **Itil Admin** has the elevated ITSM administration privileges needed to perform incident closure activities. This role supports administration of incident records and the associated process controls, including the ability to complete closure when the incident has been resolved and the required closure information is

available. ServiceNow separates the ability to work and resolve incidents from the more controlled action of formally closing them, helping preserve an auditable closure process.

The **Incident caller** can also close an incident that they submitted. This supports the self-service confirmation step: after support has resolved the issue, the caller can confirm that service has been restored and close the incident from the user-facing experience. Allowing the affected user to perform this action provides validation that the resolution meets the caller's needs before the record is finalized. These permissions align with ServiceNow's Incident Management lifecycle and its role-based access approach. See the ServiceNow Incident Management documentation and the ServiceNow roles documentation.

QUESTION NO: 51

Which of the following CANNOT be performed by public user?

A. Read Categories

B. Read Catalog Item

C. Read Catalog Tasks

D. Read Catalog Definition

ANSWER: C

Explanation:

Read Catalog Tasks cannot be performed by a public user. In ServiceNow, a public user is an unauthenticated guest context intended for limited access to content that has explicitly been made publicly available, such as eligible Service Catalog browsing information. A catalog task is a fulfillment-work record in the Catalog Task table (`sc_task`). It can contain assignment details, fulfillment instructions, work notes, requester-related information, and other operational data. Reading these records is therefore restricted through platform access controls and is intended for authenticated users with an appropriate business need, such as assigned fulfillers or users involved in the requested item's fulfillment process.

This separation supports ServiceNow's least-privilege model: public access may expose a catalog's available offerings where configured, while the internal work created to deliver an offering remains protected. ACL evaluation determines whether a user can read a table or record, and a guest/public session does not receive the access required to view catalog-task fulfillment data. ServiceNow describes ACLs and their role in controlling table and record access in its Access control rules documentation. For broader guidance on protecting application data and enforcing access, see the ServiceNow Developer access controls learning module.

QUESTION NO: 52

Which two actions can an agent perform from a major incident record? (Choose two.)

A. Create a major incident communication.

B. Create a problem record.

C. Convert the major incident into a knowledge base.

D. Change the major incident into a service offering.

ANSWER: A B

Explanation:

An agent can create a **major incident communication** to provide stakeholders with information about the outage, its impact, and restoration progress. Major incident communications help ensure that users and affected teams receive consistent updates during a high-impact interruption.

An agent can also create a **problem record** from the major incident to begin root-cause analysis after service restoration or while investigation continues. The problem record preserves the connection between the significant service interruption and the work needed to prevent recurrence. See the ServiceNow Major Incident Management documentation.

QUESTION NO: 53

In the baseline instance, which of the following states are available for incidents? (Choose all that apply.)

A. Awaiting Info

B. Open

C. Canceled

D. New

E. On Hold

ANSWER: C D E

Explanation:

Canceled, **New**, and **On Hold** are available incident states in the baseline ServiceNow Incident Management process. An incident is initially logged in the **New** state, which indicates that it has been created and awaits assessment, assignment, or active work. **On Hold** is used when progress cannot continue temporarily, such as when more information, a vendor response, or a scheduled maintenance window is needed. The selected hold reason provides additional context while the incident remains unresolved. **Canceled** is available when an incident should no longer be worked, for example when it was logged in error, is determined to be a duplicate, or is otherwise withdrawn before resolution.

These states are part of the baseline incident lifecycle, which also includes states such as In Progress, Resolved, and Closed. State values control the incident's position in that lifecycle and support reporting, automation, service-level management, and clear communication with affected users. ServiceNow describes the Incident Management process and its lifecycle behavior in its product documentation: ServiceNow Incident Management documentation. Additional implementation guidance is available in the Incident Management process documentation.

QUESTION NO: 54

What are some good practices for guiding your customers' use of Notifications?

Choose 3 answers

A. When possible, maximize the quality or email updates to customers

B. Use incident itil role template as the master template to build all other ITSM templates

C. Get input from Marketing department, regarding format of customer/caller facing notifications

D. Make sure Notification requirements and test plans are in the project scope from the start

E. Use templates to ensure consistency and ease of configuration

ANSWER: B C D

Explanation:

Using the Incident ITIL role template as the master template to build other ITSM templates provides a consistent, proven baseline for notification content and configuration. Starting from a shared ITSM-oriented template helps implementation teams apply common patterns for recipients, subject lines, message content, and branding while adapting notifications to the relevant process. ServiceNow notification records support reusable configuration and can be tailored to the record, recipients, conditions, and delivery requirements of each process.

Getting input from the Marketing department regarding the format of customer- or caller-facing notifications is also an important practice. Customer communications represent the organization, so Marketing can help align wording, voice, branding, accessibility, and communication standards with established customer-facing guidance. This produces notifications that are both operationally useful and consistent with the organization's communication approach.

Including notification requirements and test plans in project scope from the start ensures notifications are designed, configured, and validated as part of the service workflow rather than added late in the implementation. Testing should verify triggering conditions, recipients, content, links, and expected behavior for realistic scenarios. ServiceNow's notification administration documentation is available at ServiceNow Notifications documentation, and configuration guidance is available in the Notification configuration documentation.

QUESTION NO: 55

You have just released a new Change Model to the testers. Testers report they can see the old change models but cannot see the new change model on the change landing page. What could cause this?

A. Workflow has not been published

B. Testers need itil role to see me change models

C. New change models are only visible to Change Managers

D. New change model needs Active to be set to True

ANSWER: D

Explanation:

New change model needs Active to be set to True is correct because the change landing page presents change models that are available for use, and a model must be active before it is offered to applicable users. The **Active** setting is an availability control on the change model record. A model can be fully configured, including its process/workflow details, yet remain unavailable from the landing page if it was saved or released with that setting cleared. Setting the model to active makes it eligible to appear when a user opens Change and selects a model, subject to the user's existing access and any model criteria configured in the instance. Since the testers can already see older models, their ability to access the landing page and use change models has been established; the key operational difference to verify on the newly released record is its active status. After updating the setting, the administrator should save the model and have testers reload the landing page so the current list is displayed. ServiceNow describes change models as reusable templates that guide creation of change requests and are selected from the Change landing page. See the ServiceNow Change models documentation and the ServiceNow guidance for creating change models.

QUESTION NO: 56

When does the Actual End Date get auto-populated in the change request form?

A. When state changes to Review.

B. When state changes to Closed.

C. When state changes to Implement.

D. Actual End Date is never auto-populated. It should be set manually.

ANSWER: A

Explanation:

The Actual End Date is auto-populated when a change request moves to the Review state. In the standard ServiceNow Change Management lifecycle, Review follows implementation and is the point at which the implementation work is considered complete and the change enters its post-implementation evaluation stage. ServiceNow records the actual

completion timestamp at this transition, providing an objective value for measuring implementation duration, supporting reporting, and documenting the timeline used during the review process.

This behavior aligns the actual dates with the operational change lifecycle: the actual start is associated with beginning implementation, while the actual end reflects that implementation has finished and the change is ready for review. Capturing the date automatically also improves consistency across change records and avoids reliance on an agent entering a completion time manually. The value can subsequently be used in change reporting, audit evidence, and analysis of planned-versus-actual implementation performance. See the ServiceNow Change Management documentation and the Change Management process documentation for the standard lifecycle and state progression.

QUESTION NO: 57

When does a State of On Hold pause an SLA?

A. Never

B. Always

C. Depends upon the SLA conditions

ANSWER: C

Explanation:

“Depends upon the SLA conditions” is correct because setting a task record to *On Hold* does not, by itself, universally pause every ServiceNow SLA. Whether timing pauses is determined by the conditions configured on the individual SLA definition, specifically its pause condition. An SLA definition can evaluate fields on the related task, such as State, and pause its clock when the record meets the configured criteria. For example, an administrator may configure a pause condition that evaluates to true while State is On Hold, perhaps together with other requirements such as a particular hold reason. Once the record no longer meets that pause condition, the SLA can resume timing if its other lifecycle conditions continue to apply.

This makes SLA behavior configurable for different processes and commitments. Teams can choose to pause an SLA while they await customer information, a vendor response, or another legitimate dependency, while leaving the SLA active for other kinds of holds. The applicable SLA definition—not the State value alone—therefore controls whether On Hold pauses the clock. ServiceNow describes pause conditions as part of SLA definition configuration and explains that they determine when an SLA is paused. See the ServiceNow SLA workflow documentation and the ServiceNow SLA documentation.

QUESTION NO: 58

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: C D

Explanation:

Set *Enable auto closure of incidents based on Resolution date* to *Yes* so that the auto-close calculation uses the time at which the incident entered the Resolved state. This is essential when the business requirement is specifically seven days after resolution, because subsequent updates to a resolved record must not restart or extend that period. ServiceNow exposes this behavior through Incident Management properties, allowing the instance’s incident-closing process to use the resolution timestamp as its baseline.

Modify the Incident Lifecycle flow to expire after seven days so the lifecycle automation performs the transition from Resolved to Closed at the required interval. The lifecycle flow provides the automated process that evaluates and closes eligible resolved incidents; configuring its expiration duration to seven days implements the requested timing. Together, the resolution-date setting establishes the correct date basis and the lifecycle-flow expiration establishes the seven-day waiting period. This configuration is preferable to a manual UI action because it consistently applies the closure policy across qualifying incident records. See the ServiceNow Incident Management properties documentation and the Incident Lifecycle Management documentation.

QUESTION NO: 59

You can create a standard change from an incident by clicking on the "Create Standard Change" UI Action in an incident form.

A. True

B. False

ANSWER: A

Explanation:

True is correct. ServiceNow Change Management supports creating a standard change directly from an incident when the *Create Standard Change* UI action is available on the Incident form. This action opens the standard-change creation flow, where the user selects an appropriate published standard change template from the Standard Change Catalog. The resulting change record is associated with the originating incident, preserving context such as the affected service, configuration item, and incident details as applicable.

Standard changes are intended for low-risk, repeatable, and preauthorized work that follows a documented procedure. Creating one from an incident is useful when resolving the incident requires a known, routine implementation activity rather than an emergency or a uniquely assessed normal change. The UI action streamlines the handoff from incident resolution to controlled change execution while retaining traceability between the records. Availability can depend on the organization having Change Management and standard change functionality configured, suitable published templates, and the user having the required permissions. ServiceNow documents this incident-based creation process in its Change Management guidance: Create a standard change from an incident. See also the Standard changes documentation.

QUESTION NO: 60

Which three statements are correct about change sets? (Choose three.)

A. When a change set is deployed to production, by default, all unmanaged Apex tests will automatically be run.

B. There is a feature that allows the user to validate a change set before deploying it.

C. When a component is added to a change set, all dependent components automatically get added.

D. When custom fields are added to a change set, field-level security is automatically included.

E. To use change sets, at least one inbound and one outbound connection must exist.

ANSWER: A B E

Explanation:

Change sets use deployment connections between Salesforce organizations and support a controlled promotion process from a source organization to a target organization. An outbound change set is created and uploaded from the source, while the target organization receives it as an inbound change set; therefore, a usable deployment path requires the corresponding outbound and inbound connection arrangement.

Before deployment, Salesforce provides a validation capability. Validation runs the selected deployment checks, including the chosen test execution level, without committing metadata changes to the target organization. This allows an administrator to identify deployment or test failures and address them before the actual deployment window.

For a deployment to a production organization, Salesforce's default test behavior is to run local Apex tests. Local tests are the organization's unmanaged tests and exclude tests belonging to managed packages. This default helps ensure that metadata changes are verified against the target organization's custom Apex before deployment completes. These behaviors are part of Salesforce's documented change-set deployment process and production test requirements. See Deploy Changes Using Change Sets and Run Tests During a Deployment.

QUESTION NO: 61

When using Catalog Builder, what can be built using templates?

Choose 2 answers

A. Content items

B. Catalog items

C. Knowledge articles

D. Order guides

E. Record producers

ANSWER: B D

Explanation:

Catalog Builder supports template-based creation for **Catalog items** and **Order guides**. A catalog item template provides a reusable starting structure for a requestable item, allowing authors to standardize commonly needed configuration rather than repeatedly building the item setup from the beginning. This supports consistent catalog design and accelerates delivery of common employee or customer requests.

Order guides can likewise be built from templates. An order guide presents a guided ordering experience that can help users request a related set of catalog items together. Using an order-guide template supplies a structured foundation for configuring that experience, including the guide's general setup and the related request process. Templates are therefore useful where organizations want repeatable, governed patterns for both individual request offerings and bundled, guided requests.

These capabilities align with Catalog Builder's purpose of providing a guided authoring interface for Service Catalog content. For ServiceNow documentation on Catalog Builder and its supported catalog-content authoring capabilities, see ServiceNow Catalog Builder documentation and ServiceNow catalog item documentation.

QUESTION NO: 62

Which two actions could the delegated administrator complete after delegated administration is granted for a custom object? (Choose two.)

A. Add a custom picklist field to the object.

B. Change organization-wide sharing rules for the object.

C. Change the relationship with another object.

D. Create a custom tab for the object.

ANSWER: A D

Explanation:

After delegated administration is configured for a custom object, the delegated administrator can perform selected customization tasks for that object without receiving full system-administrator access. Adding a custom picklist field is supported because delegated administrators can create and manage custom fields on custom objects that are included in their delegated-administration responsibilities. A picklist is a standard custom-field type, so it can be used to provide a controlled list of values on records for the delegated object.

Creating a custom tab for the object is also an available customization action. A custom object tab gives users a navigation entry point for viewing the object's records and can be configured as part of making the delegated custom object available to the appropriate users. These capabilities let an organization distribute day-to-day ownership of a business-specific object while retaining centralized control of broader security and data-model decisions. Salesforce's delegated-administration overview and custom-object documentation describe the applicable administrative and customization capabilities: Delegated Administration and Custom Objects.

QUESTION NO: 63

Which two deployment tools can be used to deploy metadata from a Developer Edition organization to another organization? (Choose two.)

A. Data Loader

B. Salesforce Extensions for Visual Studio Code

C. Change sets

D. Ant Migration Tool

ANSWER: C D

Explanation:

Change sets and the Ant Migration Tool can both deploy Salesforce metadata from a Developer Edition organization to another organization. Change sets provide a setup-based mechanism for moving supported customizations, such as custom objects, fields, flows, and other metadata components, between organizations that have an established deployment connection. An outbound change set is created in the source organization, uploaded to the connected target organization, and then validated or deployed there.

The Ant Migration Tool supports metadata deployments through the Metadata API. It is particularly useful when a team needs a repeatable, scriptable deployment process, wants to retrieve metadata into a local project directory, or needs to automate deployments as part of a build or release process. The tool uses an Ant build file together with credentials and a package definition to retrieve or deploy selected metadata components. Salesforce documents change sets as a deployment option for connected organizations and documents Ant-based migration as a Metadata API deployment approach. See Salesforce Change Sets documentation and the Salesforce Ant Migration Tool guide.

QUESTION NO: 64

Incidents can be created and managed in the workspace, using UI layouts that are tailored to different personas, processes, and interfaces. Examples include:

• Default

• Major incidents

• Self Service

• Mobile

What are these UI layouts called in the Now Platform?

A. Form Layouts

B. Workspaces

C. Forms

D. Form Designs

E. Views

ANSWER: E

Explanation:

Views is correct because a view is a named configuration that controls how a record form is presented for a particular context, user experience, or business process. ServiceNow provides the Default view and allows administrators to create or configure additional views, such as a major-incident-focused view, a self-service-oriented view, or a mobile-specific view. Each view can expose the fields, sections, related lists, and arrangement that best support the intended persona or interface while using the same underlying record and table. For incident management, this enables agents, major incident managers, and end users to work with an interface appropriate to their responsibilities without requiring separate incident record types. Views can be selected through the form's view mechanism and are used by ServiceNow to deliver context-sensitive form experiences across platform interfaces. This distinction is important in ITSM implementation: the incident process and its data remain consistent, while the presentation can be adapted to the audience and channel. ServiceNow describes views as alternative ways to display forms and lists, with configuration available for each view. See the official ServiceNow documentation on views and the instructions for creating form views.

QUESTION NO: 65

What are three benefits of using declarative customizations over code? (Choose three.)

A. Declarative customizations cannot generate run time errors.

B. Declarative customizations will automatically update with each Salesforce release.

C. Declarative customizations do not require user testing.

D. Declarative customizations are not subject to governor limits.

E. Declarative customizations generally require less maintenance.

ANSWER: B D E

Explanation:

Declarative customizations will automatically update with each Salesforce release, are not subject to Apex governor limits, and generally require less maintenance. Declarative tools use Salesforce's platform-managed configuration rather than custom Apex code. Because Salesforce operates and upgrades the underlying platform, supported point-and-click configurations continue to run on the current release without requiring the organization to revise implementation code solely to align with platform changes. This can substantially reduce the ongoing effort associated with reviewing, testing, and maintaining custom logic.

Governor limits are resource limits that Salesforce applies to Apex transactions, such as limits on SOQL queries, DML statements, and CPU time. A declarative configuration does not itself introduce Apex code that consumes those Apex-specific limits. This makes declarative approaches especially valuable when the needed business behavior can be delivered with standard configuration capabilities. Administrators can also implement and adjust these configurations more quickly, while retaining Salesforce's managed upgrade model. Salesforce recommends using declarative functionality where it meets the requirement and reserving programmatic development for needs that cannot be addressed through configuration. See Salesforce Governor Limits and Salesforce Customization Overview.

QUESTION NO: 66

Which of the following knowledge workflows are available OOTB? (Choose all that apply.)

A. Knowledge - Approval

B. Knowledge Instant Publish

C. Knowledge - Approval Publish

D. Knowledge Instant Retire

E. Knowledge - Advanced Approval

ANSWER: B C D

Explanation:

ServiceNow provides the Knowledge Instant Publish, Knowledge - Approval Publish, and Knowledge Instant Retire workflows out of the box for common knowledge article lifecycle requirements. Knowledge Instant Publish supports publishing an article directly, without an approval activity, when the knowledge base's governance model permits immediate publication. Knowledge - Approval Publish supports a controlled publication process in which the article proceeds through approval before it is published. This enables knowledge managers to apply review and authorization requirements before content becomes available to users.

Knowledge Instant Retire supports retiring an article immediately when it is obsolete, superseded, or no longer appropriate for users. These workflows represent standard publish and retirement paths that can be selected and tailored as part of a knowledge base's article lifecycle configuration. ServiceNow Knowledge Management supports defining lifecycle processes and governance for creating, reviewing, publishing, and retiring knowledge content. See the ServiceNow Knowledge Management product page and the ServiceNow product documentation portal for Knowledge Management lifecycle and workflow configuration documentation.