

DUMPS ARENA

Salesforce Certified Agentforce Specialist

Salesforce Agentforce-Specialist

Version Demo

Total Demo Questions: 42

Total Premium Questions: 420

Buy Premium PDF

<https://dumpsarena.co>

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

Topic Break Down

Topic	No. of Questions
Topic 1, AI Agents	188
Topic 2, Prompt Engineering	107
Topic 3, Data Cloud for Agentforce	76
Topic 4, Deployment Lifecycle	34
Topic 5, Multi-Agent Interoperability	15
Total	420

QUESTION NO: 1

Choose 1 option.

Which scenario best illustrates the use of Model Context Protocol (MCP) in an enterprise AI deployment?

- A. A legal assistant agent using MCP to dynamically find a document classification API to analyze case files
- B. A customer service agent engaging another agent in real-time conversation to resolve tickets
- C. A sales agent discovering other agents' capabilities using Agent Cards

ANSWER: A

Explanation:

The correct scenario is “**A legal assistant agent using MCP to dynamically find a document classification API to analyze case files.**” Model Context Protocol is an open protocol designed to give AI applications a standardized way to connect with external systems that expose useful context, data, and tools. In an enterprise setting, an MCP client can connect to an MCP server, obtain the available tool definitions and input schemas, and invoke an appropriate tool during an agent task. This avoids requiring every model integration to be custom-built around a separate proprietary interface.

Here, the legal assistant needs document-classification functionality while analyzing case files. Using MCP to locate and use an API that provides that capability demonstrates the protocol's central value: standardized, runtime access to an external enterprise tool. The agent can use the tool's declared interface to submit relevant document content or references and incorporate the resulting classification into its workflow. This is especially useful when organizations want governed, reusable integrations between AI agents and business services. The MCP architecture and its tool-discovery model are described in the [Model Context Protocol introduction](#) and the [MCP tools specification](#).

QUESTION NO: 2

Universal Containers would like to route SMS text messages to a service rep from an Agentforce Service Agent. Which Service Channel should the company use in the flow to ensure it's routed properly?

- A. Messaging
- B. Route Work Action
- C. Live Agent

ANSWER: A

Explanation:

Messaging is the appropriate Service Channel for routing an SMS conversation from an Agentforce Service Agent to a service representative. Salesforce treats SMS as a messaging conversation, and Omni-Channel uses the Messaging service channel to identify the work type and apply routing configuration such as queues, skills, agent capacity, and presence statuses. In the flow that manages the handoff, the routing step uses the established Messaging channel configuration so that the incoming conversation is delivered as a messaging work item to an available, eligible rep.

This distinction is important for an Agentforce escalation because the handoff must preserve the customer's conversation in the messaging experience while transferring responsibility to a human agent. Configuring Messaging also aligns the channel with Salesforce's digital engagement and Omni-Channel routing model, allowing the organization to manage SMS interactions consistently with other supported messaging entry points. Salesforce's Omni-Channel learning materials describe service channels as the mechanism that defines the type of work being routed, and its Messaging documentation covers SMS as a supported messaging channel. See [Salesforce Trailhead: Omni-Channel Basics](#) and [Salesforce Help: Set Up SMS Messaging](#).

QUESTION NO: 3

Universal Containers (UC) is looking to enhance its operational efficiency. UC has recently adopted Salesforce and is considering implementing Agent to improve its processes.

What is a key reason for implementing Agent?

- A. Improving data entry and data cleansing
- B. Allowing AI to perform tasks without user interaction
- C. Streamlining workflows and automating repetitive tasks

ANSWER: C

Explanation:

Streamlining workflows and automating repetitive tasks is a key reason to implement Agentforce. Agentforce agents can be configured to take action across Salesforce using topics, instructions, and actions, allowing organizations to handle recurring business processes more efficiently. For example, an agent can help process customer requests, retrieve and summarize relevant CRM information, update records, or initiate defined flows. This reduces the manual effort required for routine work and helps employees focus on tasks that require judgment, relationship-building, or exception handling.

Operational efficiency comes not merely from generating answers, but from connecting conversational requests to governed Salesforce actions and established business processes. Administrators can use standard or custom actions, including Salesforce Flow and Apex where appropriate, to ensure the agent follows the organization's intended workflow. Agentforce's configuration model also supports clear boundaries for what the agent is intended to do, so automation can be applied consistently to common requests. Salesforce describes Agentforce as a platform for building autonomous agents that can take action across business functions, and its agent actions are designed to connect agents to the work performed in Salesforce. See [Salesforce Help: Agentforce](#) and [Salesforce Developer Documentation: Agentforce Overview](#).

QUESTION NO: 4

An Agentforce configured Data Masking within the Einstein Trust Layer.

How should the Agentforce Specialist begin validating that the correct fields are being masked?

- A. Use a Flow-based resource in Prompt Builder to debug the fields' merge values using Flow Debugger.
- B. Request the Einstein Generative AI Audit Data from the Security section of the Setup menu.
- C. Enable the collection and storage of Einstein Generative AI Audit Data on the Einstein Feedback setup page.

ANSWER: B

Explanation:

Request the Einstein Generative AI Audit Data from the Security section of the Setup menu. This is the appropriate starting point because Einstein Generative AI Audit Data provides an auditable record of generative AI requests and responses processed through the Einstein Trust Layer. The audit data enables the Agentforce Specialist to inspect how information was handled during an interaction, including the data that was masked before being sent to the large language model and restored afterward. Reviewing these records lets the specialist compare the configured masking rules with the actual request payload behavior and confirm that sensitive values from the intended fields are being protected.

This validation approach is especially useful because it relies on recorded evidence from real generative AI activity rather than assumptions about the configuration. After requesting the audit data, the specialist can download and review the available logs to verify that the expected fields and values are consistently represented as masked tokens where applicable. Salesforce documents the request and access process for this data in its [Einstein Trust Layer audit documentation](#). Additional context on how the Trust Layer protects data, including data masking, is available in [Salesforce Trusted AI](#).

QUESTION NO: 5

Universal Containers wants to utilize Agentforce for Sales to help sales reps reach their sales quotas by providing AI-generated plans containing guidance and steps for closing deals. Which feature meets this requirement?

- A. Create Account Plan
- B. Find Similar Deals
- C. Create Close Plan

ANSWER: C

Explanation:

Create Close Plan meets this requirement because it is designed to help a seller move a specific opportunity toward a successful close. Agentforce for Sales can use the opportunity's available CRM context to generate an actionable, AI-assisted plan that organizes recommended next steps and guidance for progressing the deal. This gives sales representatives a focused path for addressing deal-specific activities, engaging the appropriate stakeholders, and maintaining momentum against the opportunity's expected close date and sales goals. Because the stated need is for generated guidance and steps specifically aimed at closing deals, a close plan directly aligns the AI output to opportunity execution and quota attainment. Salesforce positions Agentforce for Sales as an AI capability embedded in sellers' workflows that can reason over trusted business data and assist with sales tasks, allowing reps to spend more time advancing customer conversations and less time assembling plans manually. Learn more through Salesforce's [Agentforce for Sales overview](#) and the official [Agentforce Trailhead learning content](#).

QUESTION NO: 6

Universal Containers has implemented an agent that answers questions based on Knowledge articles. Which topic and Agent Action will be shown in the Agent Builder?

- A. General Q & A topic and Knowledge Article Answers action.
- B. General CRM topic and Answers Questions with LLM Action.
- C. General FAQ topic and Answers Questions with Knowledge Action.

ANSWER: C

Explanation:

General FAQ topic and Answers Questions with Knowledge Action is correct because this is the standard Agentforce configuration for an agent that responds to common customer questions using Salesforce Knowledge. The General FAQ topic provides the scope and instructions for handling broad informational requests. Within that topic, the Answers Questions with Knowledge action enables the agent to search relevant published Knowledge articles and use the retrieved content to ground its response. This grounding is important: rather than relying solely on the language model's general reasoning, the agent bases its answer on organization-approved support content, helping provide responses that are accurate, relevant, and consistent with the company's documented policies and procedures. In Agent Builder, topics organize the work an agent can perform, while actions define the specific capability available to the agent within that topic. Salesforce's Agentforce learning materials describe using Knowledge-based answer actions for FAQ and self-service scenarios, and Salesforce Knowledge is designed to centralize articles that support customer and service-agent questions. See [Agentforce Service Agent Basics](#) and [Salesforce Knowledge Overview](#).

QUESTION NO: 7

Coral Cloud Resorts (CCR) uses Agentforce to assist customers with booking and service issues. CCR wants to implement a triage

process 50 that:

* High severity requests must be escalated to a human service rep.

* Lower severity requests should result in creating a support case for the guest.

The requirement is to achieve the highest reliability and determinism in the response from the agent.

Which approach should an Agentforce Specialist recommend?

A. Write the triage and routing logic in Topic Instructions using an IF, THEN, ELSE pattern: “Escalate to human service rep if the request is considered severe, otherwise create support case”.

B. Use absolute keywords like “Always” and “Never” in Topic Instructions to enforce logic, such as “Always escalate when severity is high” and “Never create a support case when severity is high”.

C. Create a custom variable severityLevel populated by a Triage action. Add filters so the “Escalate to human service rep” action only runs when severityLevel = ‘High’, and the “Create Support Case” action runs only when severityLevel != ‘High’.

ANSWER: C

Explanation:

Create a custom variable severityLevel populated by a Triage action. Add filters so the “Escalate to human service rep” action only runs when severityLevel = ‘High’, and the “Create Support Case” action runs only when severityLevel != ‘High’. This design separates the classification step from the routing step, allowing the agent to store a defined severity result and use it as an explicit condition for action availability. Once the Triage action sets the variable, action filters enforce the routing path programmatically: a high-severity result exposes only the human-escalation path, while any other result exposes the case-creation path. This produces predictable execution and supports auditing because the selected action can be traced to the recorded variable value and filter condition rather than relying solely on the model’s interpretation of prose instructions. Agentforce actions can use variables and conditional filtering to control when an action is available, which is especially appropriate for operational workflows such as service triage and escalation. Salesforce’s agent-building learning materials describe using topics, actions, and instructions together, while Flow-based actions provide structured business-process execution. See [Salesforce Trailhead: Agentforce Basics](#) and [Salesforce Trailhead: Agentforce Builder Basics](#).

QUESTION NO: 8

Universal Containers has deployed Agentforce to handle customer order tracking, returns, and loyalty support. The agent needs to balance conversational flexibility for customer inquiries with guaranteed execution of identity verification steps before accessing the account information. The development team is evaluating how to structure the agent’s instruction pattern to meet both requirements.

Which statement correctly describes hybrid reasoning in Agentforce Agent Script?

A. Hybrid reasoning uses multiple large language model (LLM) models simultaneously, with one model handling conversational responses and another model executing deterministic business logic through Flow integration.

B. Hybrid reasoning requires Canvas View for declarative instructions and Script View for procedural instructions, as each editor provides different capabilities for the respective instruction types.

C. Hybrid reasoning combines declarative natural language instructions that allow large language model (LLM) interpretation with procedural instructions using the -> prefix that enforce guaranteed execution order.

ANSWER: C

Explanation:

Hybrid reasoning combines declarative natural language instructions that allow large language model (LLM) interpretation with procedural instructions using the -> prefix that enforce guaranteed execution order. Agent Script is designed to let an agent reason conversationally where judgment and language understanding are useful, while also providing explicit procedural controls for business-critical tasks. In this scenario, the agent can interpret a customer’s request for order tracking, returns, or loyalty assistance through natural-language instructions, but identity verification must be executed consistently before any account information is retrieved or disclosed. Procedural statements prefixed with -> establish that required control flow, making the verification step deterministic rather than leaving it to probabilistic LLM interpretation. This hybrid model is valuable for enterprise agents because it preserves a natural customer experience without sacrificing compliance, security, or reliable workflow execution. It allows teams to put strict guardrails around mandatory actions and

use flexible reasoning for the parts of a conversation that can safely vary by customer wording and context. For Agentforce platform and generative AI implementation guidance, see [Salesforce Developers: Einstein Generative AI Overview](#) and [Salesforce Help: Agentforce](#).

QUESTION NO: 9

The Agentforce Specialist of Northern Trail Outfitters reviewed the organization 's data masking settings within the Configure Data Masking menu within Setup. Upon assessing all of the fields, a few additional fields

were deemed sensitive and have been masked within Einstein 's Trust Layer.

Which steps should the Agentforce Specialist take upon modifying the masked fields?

- A. Turn off the Einstein Trust Layer and turn it on again.
- B. Test and confirm that the responses generated from prompts that utilize the data and masked data do not adversely affect the quality of the generated response
- C. Turn on Einstein Feedback so that end users can report if there are any negative side effects on AI features.

ANSWER: B

Explanation:

Test and confirm that the responses generated from prompts that utilize the data and masked data do not adversely affect the quality of the generated response is correct. Data masking in the Einstein Trust Layer protects designated sensitive fields before information is sent to a large language model. Because masking changes the context available to the model, an Agentforce Specialist should validate the affected prompts after changing the masking configuration. Testing should use representative records and realistic user requests to confirm that generated responses remain useful, accurate, appropriately grounded, and able to complete the intended business task without exposing protected values.

This validation is especially important when a newly masked field previously supplied meaningful context, such as account details, contact information, or other data used in prompt instructions. The specialist can then refine prompt wording, grounding, or the masking configuration as appropriate while maintaining the organization's data-protection requirements. Salesforce describes the Trust Layer's data masking capability and its role in protecting sensitive CRM data in [Einstein Trust Layer documentation](#). Additional implementation and security context is available from [Salesforce Trusted AI](#).

QUESTION NO: 10

Universal Containers (UC) updates the instructions and actions for an Agentforce Service Agent in Agent Builder. The specialist verifies the changes in the editable agent configuration, but users on the company website continue to interact with the prior behavior.

What should the Agentforce Specialist do next?

- A. Create a new Agentforce Data Library for the updated agent.
- B. Activate the updated agent version.
- C. Create a new website deployment for each agent configuration change.

ANSWER: B

Explanation:

Activate the updated agent version. Changes made to an agent configuration are not used by deployed channels until the updated version is activated. Activation makes the revised topics, instructions, and actions available to the website deployment.

Creating a new Data Library would not publish the updated agent configuration. Changing the website deployment alone does not cause an unactivated agent version to become active.

QUESTION NO: 11

The VP of Service at Universal Containers requires a report showing weekly deflection rate trends and escalation volumes for the Agentforce Service Agent over the past 90 days.

Which approach should the Agentforce Specialist recommend?

- A.** Enable Agentforce Health Monitoring to configure escalation rate alert thresholds and export the resulting trend data into a Tableau dashboard aligned with the VP of Service's reporting requirements
- B.** Use Agent Analytics within Agentforce Observability, which provides Tableau-powered dashboards with prebuilt deflection rate, escalation volume, and abandonment metrics calculated from agent session data stored in Data 360
- C.** Query the OTEL-compliant Session Tracing Data Model in Data 360 using CRM Analytics to build a custom dashboard with deflection rate calculations and week-over-week escalation comparisons tailored to UC's reporting standards

ANSWER: B

Explanation:

Use Agent Analytics within Agentforce Observability, which provides Tableau-powered dashboards with prebuilt deflection rate, escalation volume, and abandonment metrics calculated from agent session data stored in Data 360. This is the appropriate recommendation because the request is for ongoing, historical operational insight into how the Service Agent is performing: specifically, whether customer interactions are being deflected successfully and how frequently they require escalation to a human. Agentforce Observability is designed to expose this type of agent effectiveness data from session activity, while Agent Analytics presents it in dashboards suitable for service leaders. The VP can use the available analytics to review trends over a 90-day period, identify weekly changes in containment and escalation behavior, and make data-driven service-operations decisions without first designing a bespoke telemetry model or calculation framework. The prebuilt analytics approach also keeps the reporting aligned with Agentforce's underlying session metrics and supports repeatable executive reporting as usage grows. Salesforce describes Agentforce observability and analytics capabilities in its Agentforce resources, including guidance for monitoring agent performance and session outcomes. See [Salesforce Help: Agentforce Observability](#) and [Salesforce Trailhead: Agentforce Analytics resources](#).

QUESTION NO: 12

Universal Containers implemented Agentforce for its users. One user complains that an Agent is not deleting activities from the past 7 days. What is the reason for this issue?

- A.** Agentforce does not have the permission to delete the user 's records.
- B.** Agentforce Delete Record Action permission is not associated to the user.
- C.** Agentforce does not have a standard Delete Record action.

ANSWER: C

Explanation:

Agentforce does not have a standard Delete Record action. Agentforce works through actions that are explicitly available to an agent and configured in Agentforce Builder. Although agents can use supported standard actions and custom actions to retrieve information or perform approved business operations, record deletion is not supplied as an out-of-the-box standard Agentforce action. Therefore, an agent cannot delete activity records merely because a user asks it to do so, including activities dated within the last seven days.

To support a controlled deletion use case, an organization would need to design and expose an appropriate custom action, such as a Flow or Apex-based action, and apply the required guardrails, access controls, confirmation behavior, and record-selection criteria. This approach helps ensure that destructive operations are intentional and governed by the organization's security model. Agentforce actions are the mechanism that defines what an agent is able to do; they are not inferred from a conversational request alone. See Salesforce's [Einstein Generative AI Developer Guide](#) and the [Agentforce documentation](#) for information on Agentforce capabilities and configuration.

QUESTION NO: 13

Universal Containers plans to enhance its sales team's productivity using AI. Which specific requirement necessitates the use of Prompt Builder?

- A. Creating a draft newsletter for an upcoming tradeshow.
- B. Predicting the likelihood of customers churning or discontinuing their relationship with the company.
- C. Creating an estimated Customer Lifetime Value (CLV) with historical purchase data.

ANSWER: A

Explanation:

Creating a draft newsletter for an upcoming tradeshow requires Prompt Builder because it is a generative-AI use case: the desired result is newly composed, natural-language content. Prompt Builder lets an admin create reusable prompt templates that instruct an approved large language model to produce business-specific text. A template can include merge fields and related CRM grounding data, such as account details, products, event information, or the sales representative's name, so that the newsletter draft is relevant to the intended audience rather than generic.

This capability directly supports sales productivity by giving representatives a high-quality starting draft that they can review, refine, and send through the organization's established processes. Prompt templates can also be designed for consistent messaging and can incorporate Salesforce data dynamically when invoked from supported workflows and experiences. Salesforce describes Prompt Builder as a tool for creating prompt templates that generate content using generative AI, including sales communications and other contextual text. See [Salesforce Help: Prompt Builder](#) and [Trailhead: Prompt Builder Basics](#).

QUESTION NO: 14

Universal Containers (UC) noticed an increase in customer contract cancellations in the last few months. UC is seeking ways to address this issue by implementing a proactive outreach program to

customers before they cancel their contracts and is asking the Salesforce team to provide suggestions.

Which use case functionality of Model Builder aligns with UC 's request?

- A. Product recommendation prediction
- B. Customer churn prediction
- C. Contract Renewal Date prediction

ANSWER: B

Explanation:

Customer churn prediction is the appropriate Model Builder use case because the business objective is to identify customers likely to cancel before the cancellation occurs. A churn prediction model uses relevant historical data, such as prior customer behavior, service interactions, contract information, and engagement signals, to produce a likelihood that a customer will leave. Salesforce users can use that prediction to prioritize proactive retention outreach toward customers with the highest risk, rather than waiting for a cancellation event or applying the same intervention to every customer.

This directly supports a prevention-oriented workflow: identify risk, route or surface the prediction to the appropriate team, and take timely retention action. Model Builder provides guided use cases for prediction problems, including customer churn, so it aligns both with the requested business outcome and with the intended purpose of the feature. The prediction is not merely a report of past cancellations; it is a forward-looking score that can inform actions before a contract is terminated. See Salesforce's [Model Builder use cases documentation](#) and [Model Builder overview](#) for details on creating and using these predictive models.

QUESTION NO: 15

Universal Containers (UC) wants to improve the productivity of its sales team with generative AI technology. However, UC is concerned that public AI virtual assistants lack adequate company data to general useful responses.

Which solution should UC consider?

- A. fine-tune the Einstein AI model with CBM data.
- B. Build AI model with Einstein discovery and deploy to sales users.
- C. Enable Agentforce and deploy to sales users.

ANSWER: C

Explanation:

Enable Agentforce for Sales and deploy it to sales users is correct because it provides sales teams with generative AI that can use the organization's Salesforce data and business context rather than relying solely on the broad, non-specific knowledge available to public AI assistants. Agentforce for Sales is designed to support seller productivity with contextual assistance, including account and opportunity research, meeting preparation, follow-up support, and guidance on sales activities. The agent can be grounded in trusted enterprise data and connected to Salesforce records, enabling responses that are relevant to the customer, pipeline, products, and sales process.

This approach also aligns with Salesforce's architecture for trusted AI. Salesforce's Einstein Trust Layer is intended to help protect sensitive CRM information while generative AI requests are processed, with controls such as data masking, secure retrieval, and auditability. UC can therefore give users an AI assistant that is useful in their daily Salesforce workflow while maintaining governance over company data. Agentforce can also be configured with appropriate instructions, topics, actions, and access controls so that its behavior reflects UC's sales processes and each user only receives information they are authorized to access. See [Salesforce Agentforce for Sales](#) and [Salesforce Einstein Trust Layer on Trailhead](#).

QUESTION NO: 16

Universal Containers' Agentforce Specialist suspects the Service Agent is systematically misclassifying billing dispute intents under a general inquiry topic, causing incorrect actions to execute. The administrator needs to identify this pattern across sessions without reviewing individual transcripts.

What should the specialist recommend?

- A. Enable Session Tracing in Agentforce Observability and query the Data 360 Session Tracing Data Model directly
- B. Use Agent Optimization in Agentforce Studio, which segments production sessions into moments and generates system intent clusters weekly from cross-session analysis
- C. Upload a representative set of billing dispute utterances to the Agentforce Testing Center and run a batch test comparing expected versus actual topic classifications to identify the scope of the misclassification pattern across production intents

ANSWER: B

Explanation:

Use Agent Optimization in Agentforce Studio, which segments production sessions into moments and generates system intent clusters weekly from cross-session analysis. This is designed for finding trends in how an agent behaves across real conversations at scale. Agent Optimization organizes production session data into meaningful moments and identifies system-intent patterns, allowing an administrator to see that utterances associated with billing disputes are repeatedly being grouped with a general inquiry intent. That cross-session view directly supports diagnosing a systematic classification issue without the impractical task of reading transcripts one by one. Once the recurring cluster and its associated topic behavior are visible, the team can use the insight to refine the agent's topic instructions, classifications, and action-routing design, then monitor later optimization results to validate improvement. This recommendation fits an observability and continuous-improvement workflow based on actual production interactions, rather than an isolated test run. See Salesforce's [Agent Optimization documentation](#) and the [Agentforce Observability documentation](#) for details on analyzing agent sessions and behavior.

QUESTION NO: 17

When configuring a prompt template, an Agentforce Specialist previews the results of the prompt template they 've written. They see two distinct text outputs: Resolution and Response. Which information does the Resolution text provide?

- A. It shows the full text that is sent to the Trust Layer.
- B. It shows the response from the LLM based on the sample record.
- C. It shows which sensitive data is masked before it is sent to the LLM.

ANSWER: A

Explanation:

It shows the full text that is sent to the Trust Layer. In Prompt Builder preview, Salesforce first resolves the prompt template by replacing its merge fields and grounding references with values from the selected sample record and any configured related data. The resulting resolved prompt is displayed as the Resolution. This view is valuable because it lets the specialist verify the exact instructions, context, and record-derived content assembled by the template before generative processing occurs.

That resolved content then passes through Salesforce's Einstein Trust Layer, which applies the platform's AI safety, data-protection, and governance controls as part of the generative AI request flow. Reviewing Resolution helps ensure that the template is producing the intended prompt for the chosen record, including correctly rendered variables and grounded information. The separate Response area represents the generated result after the model processes that resolved prompt. Salesforce describes the Prompt Builder workflow and testing capabilities in its [Prompt Builder Basics](#) Trailhead module, and provides an overview of the protections involved in the [Einstein Trust Layer](#).

QUESTION NO: 18

Universal Containers has deployed several specialized Agentforce Employee Agents, such as IT Support, HR Assistant, and Procurement, to assist with internal tasks. Recently, UC's help desk has reported a high volume of failed interactions because employees are frequently selecting the incorrect agent to handle their requests, for example, asking the HR Assistant agent to reset a network password. UC wants to improve the user experience, scale their deployment, and centralize control without requiring employees to guess which agent to use.

Which architectural approach should the Agentforce Specialist recommend to resolve this issue?

- A. Create a custom validation rule on the Agent Session object to prevent users from submitting prompts that do not match the selected agent's system instructions.
- B. Implement a Single Org Multi-Agent (SOMA) to act as a unified, central entry point that interprets user intent and routes the request to the appropriate specialized capabilities.
- C. Deploy a standalone Multi-Agent architecture where each specialized agent prompts the user to verify their specific department and role before proceeding with the conversation.

ANSWER: B

Explanation:

Implement a Single Org Multi-Agent (SOMA) to act as a unified, central entry point that interprets user intent and routes the request to the appropriate specialized capabilities. This architecture directly addresses the underlying problem: employees should not need to understand an organization's agent boundaries before asking for help. A central orchestrating agent can receive a natural-language request, identify its intent and required domain, then delegate work to the appropriate IT Support, HR, Procurement, or other specialized capability. For example, a request to reset a network password can be classified and routed to the IT-focused capability even when the employee begins from a single common entry point.

This model improves the employee experience by providing one consistent place to request assistance. It also supports scalable governance because the organization can add, update, and manage specialized agents while preserving the same central user experience. Agentforce supports multi-agent orchestration patterns in which an agent can invoke other agents

as actions, enabling a primary agent to coordinate specialized work rather than requiring users to choose the correct destination themselves. See Salesforce's [Multi-Agent Orchestration documentation](#) and the [Agentforce Developer Guide](#).

QUESTION NO: 19

Universal Containers wants to reduce overall customer support handling time by minimizing the time spent typing routine answers for common questions in-chat, and reducing the post-chat analysis by suggesting values for case fields. Which combination of Agentforce for Service features enables this effort?

- A. Einstein Reply Recommendations and Case Classification
- B. Einstein Reply Recommendations and Case Summaries
- C. Einstein Service Replies and Work Summaries

ANSWER: A

Explanation:

Einstein Reply Recommendations and Case Classification directly supports both parts of the service-efficiency requirement. Einstein Reply Recommendations helps service agents during a messaging or chat conversation by using the conversation context to surface relevant, ready-to-send response suggestions. Agents can review, personalize when needed, and send a recommended reply rather than repeatedly typing standard responses to common customer questions. This reduces in-chat effort while retaining agent control over the final message.

Case Classification complements this by applying AI to historical case data to predict and recommend values for configured case fields. For example, it can suggest classifications such as a case reason, type, or other supported picklist-based values based on the information captured in the interaction. These recommendations reduce the manual review and data-entry work that commonly follows a chat, improving consistency of case records as well as handling time. Together, the features address live response assistance and post-interaction case-field completion, which are the two distinct goals in the scenario. See Salesforce's [Einstein Reply Recommendations documentation](#) and [Case Classification documentation](#).

QUESTION NO: 20

In addition to Recipient and Sender, which object should An Agentforce utilize for inserting merge fields into a Sales email template prompt?

- A. Recipient Opportunities
- B. Recipient Account
- C. User Organization

ANSWER: B

Explanation:

Recipient Account is the correct object to use alongside Recipient and Sender in a Sales email template prompt. Sales outreach frequently needs company-level context in addition to details held directly on an individual contact or lead. The recipient's related account provides that context, enabling personalized content such as the company name, industry, website, account owner, or organization-specific custom fields. For example, a generated email can refer naturally to the recipient's company and tailor its language to the company's business profile.

In Prompt Builder, the Sales Email template type is designed for personalized sales communications. It supplies grounding data through standard inputs, including the recipient, sender, and the recipient's account relationship. Referencing these merge fields gives the prompt concrete CRM data rather than relying on the model to infer or invent details. This supports more relevant, accurate sales emails while keeping the template reusable for many recipients associated with different accounts.

Salesforce's Prompt Builder learning material describes using CRM record data and merge fields to ground prompt-template output: [Prompt Builder Basics](#). Salesforce also documents email template and merge-field functionality in [Email Templates](#).

QUESTION NO: 21

Universal Containers wants to incorporate CRM data as well-formatted JSON in a prompt to a large language model (LLM).

What is an important consideration for this requirement?

- A. " CRM data to JSON " checkbox must be selected when creating a prompt template.
- B. Apex code can be used to return a JSON formatted merge field.
- C. JSON format should be enabled in Prompt Builder Settings.

ANSWER: B

Explanation:

Apex code can be used to return a JSON formatted merge field. This is important when the prompt needs CRM information in a predictable, structured representation rather than as unstructured text. An Apex grounding implementation can query and shape the relevant Salesforce records, construct only the fields appropriate for the prompt, and serialize the resulting Apex objects or collections into valid JSON. The JSON string can then be supplied through an Apex merge field in the prompt template, giving the LLM clearly named properties and nested structures that are easier to interpret consistently.

Using Apex also gives the team control over the data contract sent to the model. For example, it can normalize field names, omit null or sensitive values, combine information from related records, and limit the payload to data relevant to the requested task. Salesforce's Apex `JSON.serialize` capabilities are designed for producing JSON from Apex values, while Prompt Builder supports grounding prompt templates with Apex for dynamic, custom data retrieval and formatting. The implementation should therefore ensure that the generated content is valid JSON and that the Apex logic enforces appropriate access and data-minimization practices before data is included in the prompt. See [Salesforce Apex JSON methods](#) and [Ground Prompt Templates with Apex](#).

QUESTION NO: 22

Universal Containers tests out a new Einstein Generative AI feature for its sales team to create personalized and contextualized emails for its customers. Sometimes, users find that the draft email contains placeholders for attributes that could have been derived from the recipient's contact record. What is the most likely explanation for why the draft email shows these placeholders?

- A. The user does not have permission to access the fields.
- B. The user's locale language is not supported by Prompt Builder.
- C. The user does not have Einstein Sales Emails permission assigned.

ANSWER: A

Explanation:

The user does not have permission to access the fields. Generative email drafting uses Salesforce record data to resolve the recipient attributes referenced by the underlying prompt or email template. Salesforce enforces the running user's data-access controls, including object permissions and field-level security, when it retrieves CRM data for generative AI experiences. Therefore, if a sales user cannot read a Contact field used for personalization, such as a name, title, account-related attribute, or custom field, the system cannot retrieve that value to insert into the draft. The unresolved token can consequently remain visible as a placeholder even though the recipient Contact record contains a value.

Administrators should review the affected user's profile or permission sets and confirm Read access to the Contact object and to every field referenced by the email-generation prompt. They should also validate that the prompt's recipient and related-record data sources are configured for the intended record context. This behavior is consistent with Salesforce's trust

model: generative AI capabilities are designed to honor existing CRM permissions rather than expose data that a user is not authorized to view. See Salesforce's [Prompt Builder documentation](#) and the Trailhead module on [Prompt Builder basics](#).

QUESTION NO: 23

Universal Containers is considering leveraging the Einstein Trust Layer in conjunction with Einstein Generative AI Audit Data.

Which audit data is available using the Einstein Trust Layer?

- A. Response accuracy and offensiveness score
- B. Hallucination score and bias score
- C. Masked data and toxicity score

ANSWER: C

Explanation:

Masked data and toxicity score is correct because Einstein Generative AI Audit Data records key Trust Layer activity associated with a generative AI request and response. The Trust Layer's data-masking capability identifies sensitive values in the prompt, replaces them with placeholders before information is sent to the large language model, and restores appropriate values in the returned response. Audit records can therefore include information about the masked data used in that transaction, providing traceability for privacy and compliance review.

The Trust Layer also evaluates generated output for toxicity. A toxicity score represents the safety assessment applied to generated content and can be retained in audit data so an organization can monitor AI interactions, investigate potentially unsafe responses, and support governance processes. These records help administrators understand how Trust Layer protections operated for a given request without exposing the original sensitive values to the external model.

Salesforce describes the Trust Layer as the security architecture that includes features such as data masking and toxicity detection. For additional details, see the [Salesforce Trailhead module on the Einstein Trust Layer](#) and [Salesforce Help: Einstein Generative AI Audit Data](#).

QUESTION NO: 24

Universal Containers (UC) wants to ensure the effectiveness, reliability, and trust of its agents prior to deploying them in production. UC would like to efficiently test a large and repeatable number of utterances. What should the Agentforce Specialist recommend?

- A. Leverage the Agent Large Language Model (LLM) UI and test UC 's agents with different utterances prior to activating the agent.
- B. Deploy the agent in a QA sandbox environment and review the Utterance Analysis reports to review effectiveness.
- C. Create a CSV file with UC 's test cases in Agentforce Testing Center using the testing template.

ANSWER: C

Explanation:

Create a CSV file with UC 's test cases in Agentforce Testing Center using the testing template. Agentforce Testing Center is designed for systematic, predeployment evaluation of an agent's behavior. It lets a team define a repeatable suite of test cases in a CSV template, including user utterances and expected outcomes, then run those cases at scale instead of manually entering one prompt at a time. This directly supports UC's objective to validate a large volume of realistic interactions efficiently and consistently before enabling the agent for production users.

The resulting test run provides structured evaluation data that teams can use to refine agent instructions, topics, actions, and grounding behavior. Because the same CSV-based suite can be run again after each configuration change, UC can detect regressions and measure whether changes improve results. This repeatability is particularly important for generative AI

agents, whose quality must be assessed across many representative inputs rather than through a handful of ad-hoc conversations. Salesforce positions Testing Center as the mechanism for testing and iterating on Agentforce agents with standardized test cases before release. See [Salesforce Help: Agentforce Testing Center](#) and [Trailhead: Agentforce Testing Center](#).

QUESTION NO: 25

Universal Containers (UC) is deploying several prompt templates to assist its support agents using Salesforce's standard foundation models. Leadership requires the generated responses to consistently reflect an empathetic and highly professional tone. UC only permits the use of standard foundational large language models (LLMs).

What is the most effective prompt engineering technique the Agentforce Specialist should implement in Prompt Builder to fulfill this requirement?

- A.** Configure the prompt template tone with a dataset of past interactions using different writing styles, intensifiers, and punctuation to permanently alter the LLM default tone.
- B.** Include a direct instruction asking the LLM to role-play as a specific character, for example, "Act as an empathetic customer support agent," to provide context and establish the tone.
- C.** Include multiple-choice picklist questions within the prompt template to systematically test and correct the LLM's understanding of the desired context before generating the output.

ANSWER: B

Explanation:

Include a direct instruction asking the LLM to role-play as a specific character, for example, "Act as an empathetic customer support agent," to provide context and establish the tone. Prompt Builder is designed to guide a standard foundation model's output through clear natural-language instructions, contextual grounding, and well-defined output requirements. A direct role or persona instruction establishes the perspective from which the model should respond, while explicit tone terms such as "empathetic," "professional," "respectful," and "helpful" give the model concrete behavioral guidance for every generated response. This approach is especially appropriate when UC is using standard foundation models because it controls the model's behavior at inference time through the prompt itself, without requiring any model customization or retraining. The instruction can be placed in the template's guidance section and reinforced with concise response rules, such as acknowledging the customer's concern and using professional, supportive language. Salesforce recommends providing clear, specific instructions and relevant context to help Prompt Builder produce reliable, business-appropriate results. See the Salesforce Trailhead module on [Prompt Builder Basics](#) and Salesforce's [Prompt Builder documentation](#).

QUESTION NO: 26

Universal Containers is planning a marketing email about products that most closely match a customer 's expressed interests.

What should An Agentforce recommend to generate this email?

- A.** Standard email marketing template using Apex or flows for matching interest in products
- B.** Custom sales email template which is grounded with interest and product information
- C.** Standard email draft with Einstein and choose standard email template

ANSWER: B

Explanation:

Custom sales email template which is grounded with interest and product information is the appropriate recommendation because it gives Agentforce a defined, reusable communication structure while supplying the relevant customer and product context needed to produce a personalized message. Grounding connects the generation process to trusted CRM data, such as a customer's recorded interests and the attributes of available products. This lets the generated email describe products

that align with the recipient's expressed preferences rather than relying on generic copy. A custom template also helps Universal Containers standardize essential marketing language, tone, formatting, and calls to action across generated messages, while still allowing the content to be tailored for each customer. In practice, the organization should ensure that the grounding data is current, appropriately permissioned, and sufficiently detailed for Agentforce to identify meaningful interest-to-product matches. Salesforce describes grounded prompts as a way to provide relevant business data and context to generative AI, improving the relevance of generated output. See [Salesforce Trailhead: Prompt Builder Basics](#) and [Salesforce Agentforce](#).

QUESTION NO: 27

Universal Containers is testing an agent with a "Reset Password" action restricted by the guard available when `@variables.isVerified == True`. During a single conversation turn, the large language model successfully sets the `isVerified` flag to true using `@utils.setVariables`, but fails to invoke the "Reset Password" action immediately afterward.

What must the Agentforce Specialist consider regarding how the reasoning engine evaluates action availability?

- A. The action is available from the start; the reasoning engine's intent to solve the user's issue automatically overrides the available when guard.
- B. The action becomes available immediately; available when conditions dynamically re-evaluate after every variable change within a single turn.
- C. The action remains unavailable for that turn; available when conditions control action visibility to the LLM and evaluate before the reasoning cycle begins, so the action will not be callable until the next turn.

ANSWER: C

Explanation:

The action remains unavailable for that turn; available when conditions control action visibility to the LLM and evaluate before the reasoning cycle begins, so the action will not be callable until the next turn. In Agentforce, an action's *available when* expression is a gating mechanism that determines whether the reasoning engine can see and select that action for the current cycle. The engine forms its set of available actions from the variable state present when it begins reasoning. Although `@utils.setVariables` successfully updates `@variables.isVerified`, that update does not retroactively add an action that was excluded from the action set already presented to the model. The updated value is retained in conversation state, so when the next turn starts, the availability expression can evaluate as true and expose the Reset Password action to the reasoning engine. This behavior is important when designing flows that first establish eligibility, authorization, or verification and then perform a sensitive operation: split the process across reasoning turns or otherwise ensure the required state exists before the action-selection cycle begins. Salesforce's Agentforce learning materials describe using variables to retain and manage agent state, while action configuration controls which capabilities are available to an agent. See [Agentforce Basics on Trailhead](#) and [Agentforce documentation](#).

QUESTION NO: 28

Universal Containers (UC) uses a Flex prompt template to create executive briefings. Each briefing must include only open opportunities above a specified amount and active cases created in the previous 30 days for the selected Account.

Which approach should the Agentforce Specialist use to provide this data to the prompt template?

- A. Use related-list merge fields for all Opportunities and Cases on the Account.
- B. Create a Flow resource to retrieve and prepare the filtered records.
- C. Use the Account record as the only grounding resource and instruct the model to infer the related records.

ANSWER: B

Explanation:

Create a **Flow** resource to retrieve and prepare the required Account-related records. A flow can apply the required filters, such as opportunity stage, amount, case status, and date range, then provide the resulting data to the prompt template as focused grounding context.

Related-list merge fields are useful for standard related data but do not provide the required record-selection logic in this scenario. Including all Account data would introduce irrelevant context and make the generated briefing less reliable.

QUESTION NO: 29

Universal Containers wants to use an existing prompt template inside Flow as part of automation.

- A. Invocable Apex
- B. Einstein for Flow
- C. Flow action

ANSWER: C

Explanation:

Flow action is correct because Salesforce provides a standard Flow Core Action for invoking an activated Prompt Builder template from a flow. An admin can add the Prompt Template action to a flow, select the existing template, and map the flow's available record data or variables to the template's defined inputs. When the flow runs, the action generates the prompt response and makes that output available to later flow elements, allowing the organization to incorporate AI-generated content into a declarative automation process.

This approach is designed for admins and builders who want to combine Prompt Builder with record-triggered, screen, autolaunched, or other supported flows without custom code. The template remains centrally managed in Prompt Builder, while Flow controls when it is executed and how its generated result is used, such as updating a record, creating follow-up work, or presenting content to a user. Salesforce documents prompt-template invocation as a Flow capability through its Prompt Template action. See Salesforce's [Prompt Builder Basics](#) module and the [Prompt Builder documentation](#) for the supported configuration and activation requirements.

QUESTION NO: 30

Cloud Kicks (CK) is launching a new partner portal on Experience Cloud, CK wants to provide partners with an agent that can answer questions about product specifications from the knowledge base and allow them to submit a new Lead for a potential customer they've identified. The agent must be accessible only to authenticated partner users on the portal.

Which agent type is required to meet this scenario?

- A. Sales Agent
- B. Commerce Agent
- C. Service Agent

ANSWER: C

Explanation:

Service Agent is the appropriate agent type because it is designed for customer and external-user self-service experiences, including deployment in an Experience Cloud site for authenticated users such as partners. It can use Salesforce Knowledge as a grounding source, allowing the agent to answer product-specification questions using approved knowledge-base content rather than relying only on general language-model responses. This supports a reliable portal experience in which partners can quickly locate current product information.

The agent can also be extended beyond answering questions by adding an agent action that creates a Lead. Agentforce actions allow an agent to invoke Salesforce functionality and perform business tasks, subject to the authenticated user's permissions and the organization's sharing model. Therefore, CK can provide both Knowledge-based assistance and a

guided lead-submission capability in the same authenticated partner portal experience. Salesforce documents Service Agent as an agent for service and self-service use cases and documents how agents can be deployed to Experience Cloud sites. See [Salesforce Help: Agentforce Service Agent](#) and [Salesforce Help: Deploy Agentforce in Experience Cloud](#).

QUESTION NO: 31

When configuring a file upload-based Data Library, what are the maximum allowed file sizes for text or HTML files and PDF files, respectively?

- A. Up to 100 MB for text or HTML files and up to 4 MB for PDF files
- B. Up to 50 MB for both text or HTML files and PDF files
- C. Up to 4 MB for text or HTML files and up to 100 MB for PDF files

ANSWER: C

Explanation:

Up to 4 MB for text or HTML files and up to 100 MB for PDF files is correct. In a file upload-based Data Library, Salesforce applies file-type-specific upload limits rather than one uniform limit for every supported format. Plain-text and HTML source files are limited to 4 MB, while PDF files can be uploaded at up to 100 MB. These limits matter during library planning because the uploaded files become a source for grounding an Agentforce agent's responses. Teams preparing knowledge content should therefore confirm both the format and size of each source document before uploading it, and should split or otherwise prepare oversized text or HTML content as needed. PDF documents can accommodate substantially larger source material, which can be useful for manuals, policies, and other longer reference documents. Salesforce's Agentforce learning material explains how Data Libraries provide grounding data for agents, and the Salesforce Help documentation provides the configuration guidance and supported-source requirements. See [Ground Agentforce Agents with Data Libraries](#) and [Salesforce Help: Data Libraries](#).

QUESTION NO: 32

What is the purpose of applying filters in a custom retriever configuration?

- A. Filters narrow the search results by applying up to 10 conditions based on fields defined in the search index, thereby enhancing the relevancy of the content returned.
- B. Filters automatically encrypt and mask sensitive fields in the search index to ensure that only non-confidential information is retrieved for public queries.
- C. Filters reformat and aggregate multiple documents into a single summary output to streamline and unify retriever output for more efficient and accurate AI grounding.

ANSWER: A

Explanation:

Filters narrow the search results by applying up to 10 conditions based on fields defined in the search index, thereby enhancing the relevancy of the content returned. In a custom retriever, the search index contains the fields that can be used to identify and constrain the records or chunks eligible for retrieval. Applying filter conditions lets the retriever focus on content that matches the business context of an agent interaction, such as a particular product, locale, status, knowledge category, or customer segment. This reduces irrelevant retrieved material before it is supplied as grounding context to the large language model.

More relevant grounding is important because Agentforce uses retrieved information to formulate a response that is aligned with the organization's available data. A scoped retrieval query can therefore improve response quality and help ensure that the agent uses the most appropriate content for the current task. Filters are configured as part of retrieval behavior: they refine which indexed content is searched and returned, while the custom retriever remains responsible for finding the best matching results. Salesforce describes retrievers as the mechanism for locating relevant data to ground generative AI experiences; see the [Salesforce Generative AI Developer Guide](#) and [Salesforce Trailhead's Agentforce Retriever module](#).

QUESTION NO: 33

An Agentforce is considering using a Field Generation prompt template type.

What should the Agentforce Specialist check before creating the Field Generation prompt to ensure it is possible for the field to be enabled for generative AI?

- A. That the field chosen must be a rich text field with 255 characters or more.
- B. That the org is set to API version 59 or higher
- C. That the Lightning page layout where the field will reside has been upgraded to Dynamic Forms

ANSWER: C

Explanation:

That the Lightning page layout where the field will reside has been upgraded to Dynamic Forms is correct. A Field Generation prompt template is designed to generate a value for a supported record field directly in the Lightning record experience. Before the field can be enabled for generative AI, the specialist must verify that the field is placed on a Lightning record page using Dynamic Forms. Dynamic Forms exposes record fields as individual Field Section and field components on the Lightning page, allowing Salesforce to configure and present the generative-AI field experience in the appropriate record context. This check should happen before building the template because the target field and its Lightning page placement are central to making generated content available to users where they edit the record. The specialist should therefore confirm that the relevant record page has been migrated or upgraded to Dynamic Forms and that the intended field is available there, then configure the Field Generation template for that field. Salesforce documents Field Generation templates as a Prompt Builder capability for generating content in record fields and provides guidance on configuring Dynamic Forms for Lightning record pages. See [Salesforce Help: Field Generation Prompt Templates](#) and [Salesforce Help: Dynamic Forms](#).

QUESTION NO: 34

Universal Containers (UC) is discussing its AI strategy in an agile Scrum meeting.

Which business requirement would lead An Agentforce to recommend connecting to an external foundational model via Einstein Studio (Model Builder)?

- A. UC wants to fine-tune model temperature.
- B. UC wants a model fine-tuned using company data.
- C. UC wants to change the frequency penalty of the model.

ANSWER: B

Explanation:

“UC wants a model fine-tuned using company data.” is the business need that supports using an external foundational model through Einstein Studio’s Model Builder capabilities. Organizations may have specialized domain knowledge, vocabulary, product details, policies, or historical business data that is not sufficiently represented in a general-purpose model. Connecting an approved external model gives UC a way to select a model that better fits its use case and integrate it into Salesforce’s AI workflows while maintaining appropriate governance and access controls.

Model Builder is designed for bringing models from supported external providers into the Einstein ecosystem so they can be made available for Salesforce AI use cases, including Agentforce. This is especially relevant when the organization’s strategy calls for a model customized or otherwise adapted to its proprietary business context. The requirement is therefore about model selection and company-specific customization, rather than merely tuning an inference setting. Salesforce describes the bring-your-own large language model workflow in [Bring Your Own LLM to Einstein 1 Studio](#) and provides implementation guidance in the [Salesforce Help documentation for external models](#).

QUESTION NO: 35

An administrator wants to check the response of the Flex prompt template they 've built, but the preview button is greyed out. What is the reason for this?

- A. The records related to the prompt have not been selected.
- B. The prompt has not been saved and activated,
- C. A merge field has not been inserted in the prompt.

ANSWER: A

Explanation:

The records related to the prompt have not been selected. Flex prompt templates are designed to generate a response using data supplied through their defined inputs. Unlike a template that can be evaluated entirely from fixed text, a Flex template commonly requires one or more Salesforce records as runtime context. During preview, Prompt Builder needs a selected record for each applicable input so that it can resolve record-based values, related data, and any grounding configured in the template. Until that required context is supplied, Salesforce can't construct a meaningful request for the generative AI model, so preview remains unavailable.

After choosing appropriate records in the preview context, the administrator can generate a test response and verify that the prompt is retrieving the expected CRM data and producing suitable output. This preview step is particularly important because the actual values available through the selected records can affect the completed prompt and the response. Salesforce describes how Prompt Builder templates use defined inputs and grounding to supply business context in its [Prompt Builder Basics](#) learning module. For additional product guidance, see the [Salesforce Prompt Builder documentation](#).

QUESTION NO: 36

An administrator is responsible for ensuring the security and reliability of Universal Containers' (UC) CRM data. UC needs enhanced data protection and up-to-date AI capabilities. UC also needs to include relevant

information from a Salesforce record to be merged with the prompt.

Which feature in the Einstein Trust Layer best supports UC's need?

- A. Data masking
- B. Dynamic grounding with secure data retrieval
- C. Zero-data retention policy

ANSWER: B

Explanation:

Dynamic grounding with secure data retrieval is the Einstein Trust Layer capability that meets this requirement because it securely brings relevant, current Salesforce data into the prompt context at runtime. Rather than relying only on the model's general knowledge, grounding supplies record-based business context—such as details from CRM records—so the generated response is more relevant and based on the organization's latest data. Salesforce applies its platform security controls when retrieving this information, including the user's existing data access permissions, so users receive only data they are authorized to access. The Trust Layer also protects the prompt workflow through measures such as data masking and safeguards for interactions with large language models. This combination supports UC's goals of using current CRM information while maintaining strong data protection. In Prompt Builder, grounding is specifically used to merge Salesforce data and other approved data sources into prompt templates, enabling personalized, context-aware outputs without requiring that sensitive CRM data be retained by the underlying model provider. See Salesforce's [Einstein Generative AI Trust Layer documentation](#) and its [Prompt Builder grounding documentation](#).

QUESTION NO: 37

Universal Containers (UC) wants its AI agent to return responses quickly. UC needs to optimize the retriever 's configuration to ensure minimal latency when grounding AI responses.

Which configuration aspect should UC prioritize?

- A. Configure the retriever to operate in dynamic mode so that it modifies the search Index structure at runtime.
- B. Ensure the retriever 's filters are defined to limit the scope of each search efficiently.
- C. Increase the recency bias setting for the retriever limiting scope to more recent data.

ANSWER: B

Explanation:

Ensure the retriever 's filters are defined to limit the scope of each search efficiently. is correct because retrieval is part of the agent's end-to-end response path: before the large language model can produce a grounded answer, the retriever must locate relevant information. Narrow, well-designed filters reduce the number of records or chunks considered for a request, which reduces search work and helps the agent receive grounding context sooner. Filters should reflect stable, meaningful boundaries for the use case, such as a product line, region, record type, knowledge category, or user-appropriate data scope. This also helps the retriever return a more focused context set, allowing the model to ground its response in the most applicable information without processing an unnecessarily broad corpus. In practice, UC should define filters that are selective enough to constrain retrieval while still retaining all content needed to answer expected questions. Salesforce's Agentforce grounding guidance emphasizes connecting agents to relevant enterprise data and configuring retrieval so agents can use the appropriate context. See [Salesforce Help: Grounding](#) and [Trailhead: Agentforce Data Library](#).

QUESTION NO: 38

A service manager wants to use Salesforce Prompt Builder to help agents summarize customer case notes after a support call. The summary should:

Capture the customer's issue, troubleshooting steps taken, and next actions.

Be no longer than five sentences.

Use plain language with no technical jargon.

If no next action is identified, the summary should explicitly state "No next action required."

Which key prompt constructs meet this requirement?

- A. Role, Task, LLM Clarity Score, and Format
- B. Role, Task, Token Size Limit, and Format
- C. Task, Context, Constraints, and Format

ANSWER: C

Explanation:

Task, Context, Constraints, and Format is correct because these constructs collectively translate the service manager's business requirements into a reliable Prompt Builder instruction. The task directs the model to summarize the customer's case notes. Context identifies the information that must be represented in that summary: the customer issue, troubleshooting performed, and any next actions. Supplying relevant record information as context helps ground the generated response in the actual support interaction.

Constraints define the required guardrails for the output. In this case, they include a maximum of five sentences, plain language without technical jargon, and the mandatory fallback statement, "No next action required," when there is no identified follow-up. These explicit requirements make the expected behavior testable and consistent across summaries.

Format then specifies the intended presentation of the result as a concise, readable case summary, so agents can use it directly after a call.

Salesforce Prompt Builder supports structured prompt instructions and the use of grounding data to produce relevant, business-appropriate outputs. Clear instructions, contextual inputs, and explicit output requirements are central to effective prompt design. See Salesforce's [Prompt Builder documentation](#) and the [Prompt Builder Basics Trailhead module](#).

QUESTION NO: 39

Universal Containers has successfully chunked and vectorized its unstructured meeting notes into a Data 360 search index. An Agentforce Specialist needs to connect this data to a prompt template to dynamically refine search criteria and retrieve the most relevant information.

How should the specialist achieve this?

- A. Create a Flex template referencing a data lake object (DLO).
- B. Create a Flex template referencing a retriever in the prompt template.
- C. Create a Flex template referencing the data model object (DMO).

ANSWER: B

Explanation:

Create a Flex template referencing a retriever in the prompt template. A retriever is the Prompt Builder capability designed to ground generative AI requests with relevant content from unstructured data that has already been indexed in Data 360. After meeting notes are chunked, embedded as vectors, and stored in a search index, the retriever accepts the user or agent's query, performs semantic search against that index, and returns the most relevant chunks. The prompt template can then include those retrieved results as grounding context before the large language model generates its response.

This approach supports dynamic refinement because the retriever runs at prompt-generation time rather than relying on static text in the template. The template can use the current conversational context or supplied input to form the retrieval query, allowing each request to retrieve passages that best match the question being asked. Flex prompt templates are appropriate when the specialist needs to construct prompts with custom logic and grounding resources such as retrievers. This retrieval-augmented generation pattern helps ensure agent responses are informed by the indexed meeting notes and remain contextually relevant. See Salesforce's [Retriever documentation](#) and the [Salesforce Trailhead module on retrieval-augmented generation](#).

QUESTION NO: 40

Universal Containers wants an AI agent to answer questions about warranties using unstructured data stored in Data Cloud. Results must be filterable by product line and ranked by recent updates.

- A. Use the default retriever which automatically accounts for recency ranking.
- B. Build a custom retriever in Einstein Studio with product line filters and recency ranking.
- C. Apply semantic embeddings with default metadata filters to achieve the desired result.

ANSWER: B

Explanation:

Build a custom retriever in Einstein Studio with product line filters and recency ranking. is correct because the requirement goes beyond basic retrieval from unstructured warranty content. A retriever determines how Agentforce retrieves relevant chunks from a Data Cloud search index to ground an agent response. For this use case, product line must be configured as a field that supports prefiltering, so the retrieval request can constrain results to the relevant product category before search results are returned. This helps ensure that an answer about one warranty product line is grounded only in content for that line.

The requirement to prioritize recently updated warranty information also calls for an explicit ranking configuration. Data Cloud search indexes support ranking factors, including recency, that influence the ordering of retrieved results. Creating a custom retriever in Einstein Studio lets the team refine the search criteria used by the agent and associate it with the appropriate search index, prefilter fields, and ranking behavior. This produces retrieval that is both scoped to the requested product line and ordered to favor current information—important when warranty policies or terms change over time. Salesforce describes retrievers and their role in grounding generative AI in [Learn About Retrievers](#) and documents search-index configuration concepts in [Salesforce Help: Search Indexes](#).

QUESTION NO: 41

Universal Containers (UC) is looking to improve its sales team's productivity by providing real-time insights and recommendations during customer interactions.

Why should UC consider using Agentforce Sales Agent?

- A. To track customer interactions for future analysis
- B. To automate the entire sales process for maximum efficiency
- C. To streamline the sales process and increase conversion rates

ANSWER: C

Explanation:

To streamline the sales process and increase conversion rates is correct because Agentforce for Sales is designed to help sales teams work more efficiently with AI-driven assistance grounded in CRM data and the organization's defined business context. Sales agents can support sellers throughout the sales lifecycle by surfacing relevant information, helping with prospecting and account research, preparing for customer engagement, and handling routine sales work. This reduces time spent on manual, administrative activities and enables representatives to focus attention on the highest-value customer conversations and next steps.

For UC's stated need, real-time insights and recommendations are especially valuable because they help sellers make better-informed decisions while engaging customers, rather than requiring them to search across records or defer follow-up work. By prioritizing useful actions and helping teams respond with relevant context, Agentforce can improve seller productivity, create more consistent engagement, and support progress toward successful deal outcomes. Salesforce describes Agentforce for Sales as providing autonomous and assistive capabilities that help sales organizations scale productivity and engage buyers effectively. See [Salesforce Agentforce for Sales](#) and [Salesforce Help: Agentforce](#).

QUESTION NO: 42

Choose 1 option.

Coral Cloud Resorts is uploading thousands of new HTML knowledge articles files for a resort launch.

To ensure Agentforce retrieves accurate responses quickly, which chunking strategy should be used when creating a new index?

- A. Semantic-based passage extraction
- B. Conversation-based chunking
- C. Section-aware chunking

ANSWER: C

Explanation:

Section-aware chunking is the appropriate strategy for a large collection of HTML knowledge articles because HTML is inherently structured. Article headings, subsections, paragraphs, lists, and other markup communicate which content belongs together. A section-aware approach uses that structure to create retrieval units that retain the relationship between a

heading and its supporting details. This gives Agentforce more focused, meaningful chunks to retrieve when grounding an answer.

For resort-launch content, users might ask about a specific property amenity, booking policy, opening date, or guest service. Preserving section boundaries helps ensure that retrieved context contains the relevant topic and its associated details rather than a fragment split across unrelated portions of an article. Well-bounded chunks also improve retrieval efficiency because the search index can return smaller, contextually relevant passages for the agent to use.

Chunking is a key part of preparing unstructured content for retrieval-augmented generation: content is segmented, indexed as embeddings, and retrieved as grounding context for generative responses. Salesforce describes grounding Agentforce responses with enterprise data and retrieval capabilities in its [Agentforce overview](#) and provides implementation guidance in the [Einstein Generative AI developer documentation](#).