UiPath-AAPv1 Dumps

Shared by Roman on 11-11-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

What limitations apply when using Integration Service activities in UiPath agents, and how can files be handled in agent automations?

Options:

A- UiPath agents don't support Integration Service triggers, mandatory file I/O activities, or cascading fields. Files are best handled by calling RPA workflows as steps or tools.
B- File handling is only possible in standalone RPA workflows outside of agents, with no integration or data-passing possibilities.
C- Agents only reliably support very basic Integration Service activities with no inputs and no outputs. Anything involving files, triggers, or data exchange is usually not supported and severely limits agent capabilities.
D- All Integration Service activities, including triggers and file-dependent activities, are supported in agents without limitations.

Answer:

A

Explanation:

Option A most closely describes UiPath's Integration Service activity limitations and the recommended approach for file-processing operations.

UiPath Agents support Integration Service activities as tools, enabling governed API calls to enterprise applications through authenticated connections.

However, not every activity type is supported directly.

UiPath's October 2025 documentation specifies that API-based Integration Service activities are supported, while triggers and other non-API activity types are excluded.

Activities requiring mandatory file inputs or outputs are also excluded from direct agent activity-tool configuration. Activities containing optional file fields may remain available, but those specific file fields are omitted.

File-processing requirements can be addressed by invoking suitable RPA workflows as agent tools.

Current UiPath documentation additionally describes file-capable tools such as Analyze Files and Intelligent Xtraction and Processing (IXP), so file support is not limited exclusively to conventional

RPA.

Option A requires an important correction concerning cascading fields. UiPath does not prohibit all cascading fields. It specifically prevents them from using the Prompt input method because dependent field structures cannot be dynamically reconstructed by the agent at runtime.

Options B and C incorrectly restrict agent integration capabilities, while option D ignores documented limitations.

Therefore, A is the intended answer, with the cascading-field qualification.

Official UiPath references/topics: Agents --- October 2025 Release Notes --- Expanded Integration Service Activities Support; Activities; Working with Files.

Question 2

Question Type: MultipleChoice

An agent consistently struggles to generate accurate responses despite having a well-structured prompt.

Which evaluation strategy would best isolate whether the problem lies in prompt clarity or tool invocation logic?

Options:

A- Use exact match evaluators across multiple evaluation sets and compare scores
B- Increase the number of evaluation sets and rerun the agent with randomized tool sequences
C- Apply LLM-as-a-Judge evaluators to assess semantic alignment between prompt intent and output
D- Switch to JSON similarity evaluators and disable prompt scoring

Answer:

C

Explanation:

Option C is the strongest listed choice because LLM-as-a-Judge evaluation can determine whether generated responses satisfy the semantic intent of the agent's instructions.

UiPath's Semantic Similarity evaluator compares actual and expected responses according to meaning rather than requiring identical text.

This helps identify cases where the agent produces fluent responses that do not satisfy the intended business objective.

However, semantic evaluation alone cannot conclusively distinguish poor prompt interpretation from incorrect tool invocation.

For example, an agent may misunderstand instructions and select the wrong tool. Alternatively, it may select the correct tool but supply invalid arguments or misinterpret the returned result.

Both situations can produce low semantic evaluation scores.

Therefore, a rigorous investigation should combine output-oriented evaluation with execution traces and tool-call evaluators.

UiPath supports inspecting recorded tool arguments, outputs, execution order, and associated errors.

Runtime evaluation capabilities also include Tool Call Arguments, Tool Call Count, Tool Call Order, and Tool Call Output evaluators.

Options A and D rely on deterministic output comparison without directly identifying behavioral causes. Option B introduces artificial execution variability rather than isolating the defect.

Option C is the intended answer, but complete diagnostic isolation requires additional trajectory and tool-execution evidence.

Official UiPath references/topics: Agents --- Evaluations --- Semantic Similarity; Managing Runtime Evaluations --- Tool Call Evaluators; Agent Traces.

Question 3

Question Type: MultipleChoice

An agent extracts customer email addresses from support tickets and must check that each extracted value exactly matches the format "" without variations.

Which evaluator type is best suited to this validation requirement?

Options:

A- Model-graded evaluator, because it can assess whether the extracted value structurally resembles an email address

B- Deterministic evaluator, because it performs rule-based precise pattern matching without requiring semantic interpretation

C- LLM-as-a-Judge evaluator, because it can understand the semantic intent behind the email

format requirement
D- Semantic Similarity evaluator, because it can compare the extracted email against expected pattern variations

Answer:

B

Explanation:

Option B is the intended answer because strict format validation requires deterministic assessment rather than semantic interpretation.

UiPath deterministic evaluators use predefined comparison rules that produce consistent outcomes for identical inputs.

These evaluators are appropriate when output requirements are precisely defined and the result must not depend on subjective language-model judgments.

For instance, an Exact Match evaluator can verify that an extracted value equals a specific expected string.

When validating arbitrary email addresses against a required format, however, the implementation must distinguish exact-value comparison from pattern validation.

A regular expression or equivalent deterministic validator can enforce requirements such as permitted characters, mandatory separators, domain structure, and whether the entire input matches the specified pattern.

An LLM-as-a-Judge evaluator may recognize that a value resembles an email address, but semantic assessment does not provide the same guarantees as a deterministic syntax check.

Options A, C, and D therefore introduce unnecessary interpretation and potential scoring variability.

The original question contains an additional technical problem: the required format is represented by empty quotation marks. Without the actual format specification, the exact validation rule cannot be constructed.

Furthermore, UiPath's built-in Exact Match evaluator is not inherently a general-purpose regular-expression validator.

Official UiPath references/topics: Agents --- Evaluations --- Deterministic Evaluators and Exact Match.

Question 4

Question Type: MultipleChoice

An organization is designing a customer-facing service agent using the agent story blueprint format.

To align with UiPath agent components, which story structure best reflects proper instruction and interaction design?

Options:

A- Specify the agent department and reporting line, define desired customer outcomes, reference historical feedback as context, integrate chat interfaces, and measure performance using satisfaction surveys alone.
B- Specify the agent role and operating environment, define explicit behavioral instructions in the prompt, identify relevant contextual sources, integrate appropriate tools, and define measurable success criteria with escalation paths.
C- Specify the automation goal and system owner, define high-level service objectives, reference general organizational policies as context, integrate notification channels, and measure performance using system uptime metrics.
D- Specify the process category and compliance scope, define expected efficiency targets, reference archived cases as context, integrate analytics dashboards, and measure performance using quarterly audit reviews.

Answer:

B

Explanation:

Option B provides the most complete agent story structure because it aligns with UiPath's principal agent design components and operational recommendations.

An agent story should begin with a defined role and operating environment, establishing what the agent is responsible for and the business context in which it operates.

The prompt should specify the agent's objectives, behavioral instructions, constraints, task sequence, and expected output structure.

Context identifies the information sources required for informed decisions. UiPath Context Grounding can provide access to relevant enterprise documents and knowledge repositories.

Tools define how the agent interacts with external systems, executes automations, or retrieves

operational information.

Measurable success criteria establish how performance will be evaluated, including accuracy, response consistency, completion rates, and adherence to business requirements.

Escalation paths identify circumstances requiring human validation, approval, or exception handling.

Option A omits essential execution and governance components. Option C lacks detailed agent instructions and relies on incomplete performance indicators. Option D focuses on administrative and analytical characteristics without sufficiently defining agent behavior.

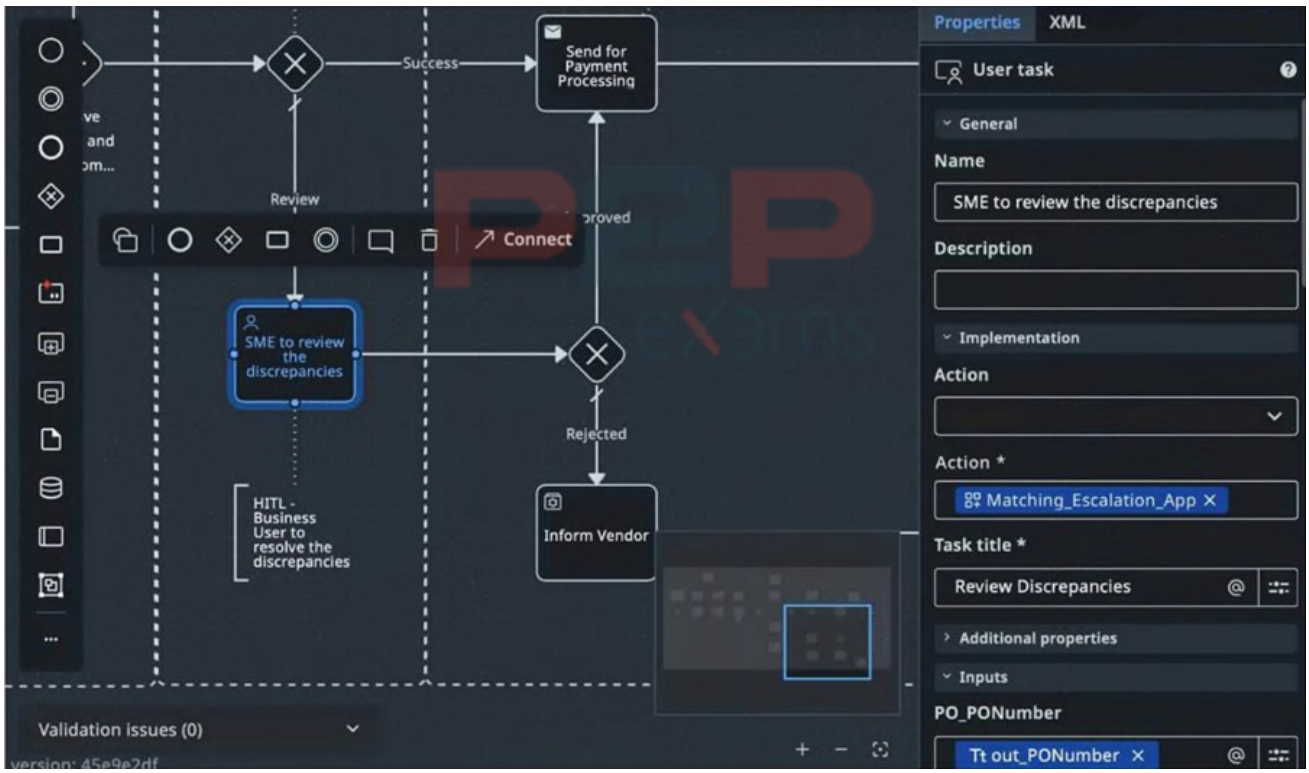
A well-designed agent story connects responsibilities, instructions, knowledge, execution capabilities, performance measurements, and human oversight into a coherent operational design.

Official UiPath references/topics: Agents --- Best Practices for Building Agents; Escalations; Contexts.

Question 5

Question Type: MultipleChoice

Given the following BPMN for the "SME to review the discrepancies" task, which option should be selected in the "Action" dropdown?



Original exhibit --- Question 56

Options:

A- Create action app task
B- Create validation app task
C- Create and wait for action app task
D- Create and wait for validation app task

Answer:

A

Explanation:

Option A identifies the documented implementation action for creating a human review task within a UiPath Maestro BPMN process.

The exhibit shows a User task named 'SME to review the discrepancies' with an Action App selected in its properties panel.

In Maestro, a User task represents an activity requiring human participation rather than autonomous execution by an AI agent or robotic automation.

The Create action app task implementation integrates UiPath Action Apps and Action Center to provide a structured interface through which an assigned user can review business information and submit a decision.

In this scenario, the subject matter expert reviews discrepancies identified during invoice and purchase-order matching.

The Action App receives relevant process variables, presents the discrepancies, and collects the reviewer's decision.

The BPMN process can subsequently route execution through approval or rejection paths according to the submitted outcome.

Option B introduces a validation-specific action not represented by the standard task implementation shown. Options C and D introduce alternative action names that do not match the documented selection for this task.

UiPath's official Maestro implementation examples use Create action app task for human approval and discrepancy-resolution activities.

The appropriate choice is therefore A.

Official UiPath references/topics: Maestro Orchestrate --- Tasks; Implementing a Complex Process; Extracting and Validating Documents.

Question 6

Question Type: MultipleChoice

What minimum number of evaluations is considered a good benchmark for a simple agent?

Options:

A- 20
B- 30
C- 50
D- 100

Answer:

B

Explanation:

Option B matches UiPath's documented evaluation benchmark for agents with relatively simple responsibilities and limited execution complexity.

UiPath recommends approximately 30 evaluations distributed across one to three evaluation sets for a simple agent.

These evaluations should represent the agent's expected operational conditions, including normal inputs, boundary cases, invalid information, and relevant error scenarios.

The benchmark is intended to encourage sufficiently broad testing rather than repeated execution of identical successful examples.

For example, a customer-intent classification agent might require evaluation cases covering different request categories, ambiguous wording, incomplete messages, spelling variations, and unsupported requests.

A properly designed evaluation set should expose weaknesses in the agent's instructions, reasoning behavior, and output handling.

For more complex agents, UiPath recommends substantially greater evaluation coverage, potentially double the simple-agent benchmark or more.

Complexity factors include the number of input arguments, output structure, tool usage, decision branches, and different user scenarios.

Option A falls below UiPath's stated benchmark. Options C and D exceed it but are not the specified recommendation for a simple agent.

Thirty evaluations should be interpreted as a practical starting benchmark, not proof of production readiness. Coverage quality, business risk, and evaluation results remain more important than the raw test count.

Official UiPath references/topics: Agents --- Evaluations --- Number of Evaluations, Evaluation Sets and Coverage Principles.

Question 7

Question Type: MultipleChoice

Which search strategy should you select when your knowledge base contains CSV-formatted tabular data?

Options:

A- Static
B- Structured
C- Semantic
D- DeepRAG

Answer:

B

Explanation:

Option B correctly identifies the Structured search strategy for querying CSV-formatted tabular information through UiPath Context Grounding.

Context Grounding supports several retrieval strategies intended for different data structures and information-processing requirements.

Semantic search retrieves relevant content using similarity-based matching. It is particularly useful for unstructured documents and natural-language knowledge retrieval.

Structured search is designed for tabular datasets where information is organized into rows and columns.

UiPath explicitly documents CSV support for the Structured search strategy.

For example, an agent may need to retrieve records from a CSV containing customer identifiers, invoice amounts, payment statuses, and transaction dates.

Structured querying provides a more appropriate mechanism for retrieving information based on these defined fields than relying exclusively on semantic similarity between textual passages.

DeepRAG is intended for more extensive document synthesis and reasoning, particularly where information must be combined across multiple documents.

Option A is not the documented structured retrieval strategy. Option C is less appropriate when the task requires querying tabular records by their fields. Option D is designed for a different class of knowledge-processing operations.

The Structured strategy should therefore be selected when the primary knowledge source is CSV-formatted data requiring structured queries.

Official UiPath references/topics: Agents --- Contexts --- Search Strategy: Structured; Automation Suite 2025.10 --- Context Grounding.

Question 8

Question Type: MultipleChoice

How can an agent with Escalations be evaluated?

Options:

A- Escalations will not execute in Evaluations.
B- It can only be evaluated manually.
C- Escalations must be simulated.
D- Escalations will be automatically mocked.

Answer:

C

Explanation:

Option C is the intended selection because escalation simulation is a recommended method of evaluating agent behavior without requiring live human intervention.

UiPath Agents use escalations to request approval, validation, or assistance from human reviewers through Action Apps and Action Center.

During evaluation, an agent may encounter a scenario requiring an escalation. Allowing the actual escalation to execute can create a human task and introduce external dependencies into the test.

UiPath supports simulation of escalation behavior so developers can supply predefined or generated responses without waiting for a reviewer.

A simulated response might represent approval, rejection, or a request for additional information. The developer can then examine whether the agent continues correctly after receiving that result.

Simulation improves repeatability, reduces operational side effects, and supports automated regression testing.

However, current UiPath documentation permits both real and simulated executions within evaluation sets. Runtime evaluations can invoke live escalations when those components are not simulated.

Consequently, option C is not universally correct as written. Option A incorrectly states that escalations cannot execute, option B excludes automated evaluation, and option D incorrectly assumes that mocking occurs automatically.

The accurate recommendation is to simulate escalations when controlled, side-effect-free evaluation is required.

Official UiPath references/topics: Agents --- Evaluations --- Configuring Simulations; Managing Runtime Evaluations.

Question 9

Question Type: MultipleChoice

You want to send an email within a BPMN.

Which Action should you select inside the task to allow you to send the email?

Options:

A- Execute script
B- Execute connector activity
C- Wait for connector event
D- Create action app task

Answer:

B

Explanation:

Option B identifies the correct Maestro implementation action for sending an email through an Integration Service connector.

UiPath Maestro uses Business Process Model and Notation (BPMN) to coordinate automation activities, enterprise applications, AI agents, and human tasks.

Within a BPMN process, a Send task can communicate with an external application using the Execute connector activity implementation.

This action invokes an Integration Service activity, such as a supported Microsoft Outlook or Gmail email-sending operation.

The developer selects the connector, configures the authenticated connection, chooses the required activity, and maps process variables to the activity's input fields.

Typical email inputs include the recipient address, subject, and message body. Relevant output values can also be mapped to process variables when supported by the activity.

Option A executes custom scripts and is not the standard connector-based approach. Option C waits for an external event rather than actively sending an email. Option D creates a human task through Action Center and does not itself perform the requested email operation.

The Execute connector activity option allows the BPMN process to interact with external systems while preserving centralized orchestration and execution traceability.

Official UiPath references/topics: Maestro Orchestrate --- Send Task and Execute Connector Activity.

To Get Premium Files for UiPath-AAPv1 Visit
https://www.p2pexams.com/products/uipath-aapv1

For More Free Questions Visit
https://www.p2pexams.com/uipath/pdf/uipath-aapv1

