

Valid UiPath-AAAv1 Exam Answers - UiPath-AAAv1 Exam Forum

UiPath Advanced Certification Exam Questions and Correct

Answers (2024 / 2025) (Verified Answers)

1. A developer needs to read data from invoices and then input the extracted data into another system. Each invoice can be processed separately and the order in which they are processed does not matter.

Which type of design is best-suited for this

automation? Iterative Process

Linear Process

Library

Transactional Process: Transactional Process

2. Which state and file in the RE Framework in the Config.xlsx file read?

*State: Initialization

1 / 11

P.S. Free 2026 UiPath UiPath-AAAv1 dumps are available on Google Drive shared by DumpsValid:
<https://drive.google.com/open?id=13Y3MEKN80uOSCSCWc-OUtOBLKOa9nCw>

Facing the incoming UiPath UiPath-AAAv1 Exam, you may feel stained and anxious, suspicious whether you could pass the exam smoothly and successfully. Actually, you must not impoverish your ambition. Our suggestions are never boggle at difficulties. It is your right time to make your mark.

UiPath UiPath-AAAv1 Exam Syllabus Topics:

Section	Objectives
Agentic Evaluations & Governance	- Guardrails and validation logic - LLM-as-a-judge metrics
Agent Blueprint Design	- Agent architecture design - Workflow decomposition
Context Grounding (RAG)	- Data grounding strategies - Retrieval-Augmented Generation concepts
Escalations & Human-in-the-Loop	- Exception handling and escalation patterns - Action Center workflows
Agent Discovery & Process Assessment	- Process suitability for agentic automation - Identifying automation opportunities

Agentic Orchestration (Maestro)	- Workflow orchestration with agents - BPMN-based process design
UiPath Platform Components	- Agent Builder and Orchestrator basics - Studio Web and Autopilot
Agentic AI Fundamentals	- Agentic automation concepts - AI agents vs rule-based automation
Prompt Engineering for Agents	- Zero-shot and few-shot prompting - System prompts and constraints
Autopilot for Everyone	- AI-assisted automation building - Use cases and capabilities

>> Valid UiPath-AAAv1 Exam Answers <<

Free PDF Quiz UiPath - UiPath-AAAv1 –Reliable Valid Exam Answers

Passing UiPath-AAAv1 Certification Exam is not an easy task? Choosing DumpsValid UiPath-AAAv1 exam training materials, passing UiPath-AAAv1 exam is quite possible. DumpsValid's UiPath-AAAv1 exam training materials is the highly certified IT professionals' collection of experience and innovation results in this field, and have absolute authority. You won't regret to choose DumpsValid.

UiPath Certified Professional Agentic Automation Associate (UiAAA) Sample Questions (Q44-Q49):

NEW QUESTION # 44

An agent uses Web Search, Slack integration, and a custom process to resolve IT support tickets. The agent must:

- * Retrieve relevant troubleshooting steps from the web.
- * Notify the user via Slack if a solution is found.
- * Escalate unresolved tickets via a custom process.

Which evaluation strategy ensures comprehensive coverage while avoiding redundancy?

- A. Use random input sampling across all tools and rely on the default "LLM-as-a-Judge" assertion.
- B. Create more than 30 evaluations for Slack notifications, more than 30 for web searches, and more than 30 for escalation processes.
- C. Group evaluations into sets: Valid web results triggering Slack notifications, Invalid web results triggering escalations, Edge cases.
- D. Create 30 evaluations for Slack notifications, 30 for web searches, and 30 for escalation processes.

Answer: C

Explanation:

C is correct - UiPath recommends structuring agent evaluations around functional sets that align with expected behavior and edge conditions. This strategy:

- * Validates end-to-end logic, not just isolated tool usage
- * Helps assess whether tool combinations work as designed
- * Supports traceable diagnosis of failures or regressions

In this scenario:

- * Set 1: Valid Web Search results#Slack notification (success path)
- * Set 2: Failed/irrelevant Web Search#Escalation (fallback path)
- * Set 3: Edge cases (e.g., ambiguous input, multiple valid matches)

This avoids the redundancy and volume bloat seen in options B and D.

Option A is too loose - relying solely on random inputs and "LLM-as-a-Judge" introduces risk of incomplete testing.

Grouping by real-world interaction patterns mirrors how agents behave in production. It ensures high coverage while keeping evaluation efficient, consistent, and tightly aligned with business logic.

NEW QUESTION # 45

What steps must be completed when creating evaluations from scratch for a new evaluation set in UiPath?

- A. Once the evaluation set is created, all included evaluations are automatically scored based only on input values and expected outputs.
- **B. Add a name to the evaluation set, provide input values and expected output, save each evaluation, and assign evaluators before running the evaluation set.**
- C. Assign evaluators immediately after creating the new evaluation set name, then configure inputs and expected outputs later.
- D. The evaluation set can only be created using imported JSON data from previous evaluations of other agents.

Answer: B

Explanation:

Is correct - creating a new evaluation set in UiPath involves a multi-step process designed to enable qualitative and quantitative review of agent behavior.

Steps include:

- * Naming the evaluation set
- * Adding input prompts and expected outputs
- * Saving each test item (often called "evaluations")
- * Assigning evaluators, who will manually or automatically score the results This process enables testable, repeatable evaluation of agent behavior before deployment - ensuring the model produces correct, useful, and safe outputs.

Options A and C are incorrect:

- * A reverses the order: inputs and expected outputs are needed before evaluators.
- * C is false - evaluation sets can be built from scratch. D implies scoring is automatic, but human reviewers or comparison logic are often required for nuanced evaluations.

This aligns with UiPath's best practices in agent validation and post-deployment assurance.

NEW QUESTION # 46

A developer is implementing a few-shot structured prompt for an email classification task. The prompt includes examples of email subjects labeled with their respective classifications, such as "Spam" or "Work." What is the most important aspect to consider when selecting examples for the prompt?

- A. Always use more than 10 examples, regardless of task complexity.
- **B. Choose examples that are diverse, relevant, and typical of the task's expected input.**
- C. Use random and unrelated examples to test the prompt's robustness.
- D. Include examples with intentionally incorrect labels to improve training.

Answer: B

Explanation:

The correct answer is C - the most critical aspect of designing a few-shot prompt in UiPath's LLM-driven agent framework is selecting examples that are diverse, representative, and relevant to the actual data the agent will encounter in production.

In a few-shot structured prompt, examples are used to demonstrate a pattern the model should follow.

UiPath recommends:

- * Using realistic examples from actual user inputs or support tickets
- * Covering edge cases or variations in phrasing and tone
- * Matching the desired output structure exactly (e.g., Input: ..., Output: ...) These patterns help the LLM infer the task correctly and maintain consistency, especially when processing unstructured inputs like email subjects.

Option A is incorrect - introducing incorrect labels degrades performance and adds confusion.

B is wrong - the number of examples depends on the task complexity and token budget. Sometimes 3-5 is ideal.

D undermines task alignment - random examples reduce accuracy and coherence.

UiPath's Prompt Engineering best practices prioritize grounded, contextually rich inputs, particularly when automating classification tasks like spam detection, triage, or intent recognition. High-quality, task-aligned examples lead to more reliable, human-like agents.

NEW QUESTION # 47

Which of the following best describes how agents handle dynamic environments?

- A. Agents rely solely on static rules without contextual learning.
- B. Agents require complete human assistance whenever processes change.
- **C. Agents adapt to changing conditions by learning.**
- D. Agents fail to execute tasks when information or processes change.

Answer: C

Explanation:

Is correct - one of the defining strengths of UiPath's agentic automation is the ability for agents to adapt to dynamic environments using LLMs and contextual grounding.

Agents differ from traditional RPA bots in that they:

- * Interpret natural language
- * Reason across structured and unstructured data
- * Adjust outputs based on real-time context, grounding, and updated knowledge When processes change - such as updates to escalation rules, variations in incoming requests, or new product names - agents can adjust without reprogramming, thanks to:
- * Flexible prompts
- * Grounded context from indexes or memory
- * Few-shot or zero-shot inference capabilities

This adaptability makes agents ideal for scenarios like email triage, customer service, or knowledge work, where inputs and conditions vary.

Option A and D falsely suggest agents are rigid or fully dependent on human intervention.

Option C applies to classic RPA bots - not LLM-powered agents.

While agents don't "learn" in the ML retraining sense during execution, they dynamically interpret and adapt within the context of each session - a key feature enabled by UiPath's Autopilot™, Context Grounding, and agent memory frameworks.

This flexibility is foundational to deploying agents in environments where rules evolve, data flows shift, or human-like understanding is needed.

NEW QUESTION # 48

When passing runtime data into an Agent, which approach ensures the input argument is actually available inside the user prompt at execution time?

- A. Declare the argument in the system prompt; any text surrounded by angle brackets (e.g., <CUSTOMER_EMAIL>) will be substituted automatically.
- B. Simply mention the variable name in plain prose-the Agent will infer the value from the workflow without special syntax.
- C. Create the argument in Data Manager and reference it verbatim inside double curly braces, e.g., {{CUSTOMER_EMAIL}}, so the name matches exactly.
- D. Use single braces like {CUSTOMER_EMAIL}, because the platform automatically normalizes the identifier.

Answer: C

Explanation:

Is correct - to pass runtime values into an agent's prompt in UiPath, you must:

- * Declare the variable in Data Manager
- * Reference it inside the user/system prompt using double curly braces, e.g., {{CUSTOMER_EMAIL}} This ensures the platform can:
 - * Substitute values at runtime
 - * Maintain traceability between arguments and prompts
 - * Provide context grounding for the LLM

Option A is incorrect - angle brackets are not used for substitution.

C is wrong - single braces {} are not valid for UiPath's binding syntax.

D is unreliable - LLMs do not infer values from prose without structured substitution.

This technique ensures consistent parameter injection for context-aware agent behavior.

NEW QUESTION # 49

.....

Our company has collected the frequent-tested knowledge into our practice materials for your reference according to our experts' years of diligent work. So our UiPath-AAA v1 exam materials are a triumph of their endeavor. By resorting to our UiPath-AAA v1 Practice Guide, we can absolutely reap more than you have imagined before. We have clear data collected from customers who chose our UiPath-AAA v1 training engine, the passing rate is 98-100 percent.

UiPath-AAA v1 Exam Forum: <https://www.dumpsvalid.com/UiPath-AAA v1-still-valid-exam.html>

- Vce UiPath-AAA v1 Files ☐ UiPath-AAA v1 Preparation Store ☐ Reliable UiPath-AAA v1 Dumps Questions ☐

[illegible]

BTW, DOWNLOAD part of DumpsValid UiPath-AAA v1 dumps from Cloud Storage: <https://drive.google.com/open?id=13Y3MEKN80uOSCSCWc-OUtOBjLKOa9nCw>