IT-DUMPS Q&A

Accurate study guides, High passing rate!
IT-dumps provides update free of charge in one year!

Exam : **UIPATH-ADPV1**

Title : UiPath (ADPv1) Automation
Developer Professional
Exam

Version : DEMO

1.What are the primary functions of the UiPath Integration Service?

A. Enables automation with a library of connectors, manages connections easily with standardized authentication, kicks off automations with server-side triggers or events, provides curated activities and events, simplifies automation design.

B. Automates UI design, manages API connections, provides limited activities and events, simplifies automation design.

C. Enables automation with API integration, manages connections with user-provided authentication, kicks off automations based on application-specific triggers, simplifies automation design with the help of third-party libraries.

D. Enables automation with UI components, manages API keys, kicks off automations with client-side triggers, provides curated events.

Answer: A

Explanation:

UiPath Integration Service is a component of the UiPath Platform that allows users to automate third-party applications using both UI and API integration. It has the following primary functions1:

Enables automation with a library of connectors: Connectors are pre-built API integrations that provide a consistent developer experience and simplify the integration process. Users can browse and import connectors from the Connector Catalog or the UiPath Marketplace, or create their own connectors using the Connector Builder (Preview).

Manages connections easily with standardized authentication: Users can create and manage secure connections to various applications using standardized authentication methods, such as OAuth 2.0, Basic, API Key, etc. Users can also share connections with other users, groups, or automations within the UiPath Platform.

Kicks off automations with server-side triggers or events: Users can trigger automations based on events from external applications, such as webhooks, email, or schedules. Users can also configure parameters and filters for the events to customize the automation logic.

Provides curated activities and events: Users can access a set of curated activities and events that are specific to each connector and application. These activities and events can be used in UiPath Studio or UiPath StudioX to design and execute automations.

Simplifies automation design: Users can leverage the UiPath Integration Service Designer to create and test integrations in a graphical interface. Users can also import common API formats, such as Swagger, YAML, or Postman collections, to generate connectors automatically.

Reference: 1: Integration Service - Introduction - UiPath Documentation Portal

2.What happens when closing a Remote Debugging Connection while a debugging execution is in progress?

A. The remote robot continues the execution after Studio closes the connection.

B. Debugging execution ends in exception, then the connection is closed.

C. Debugging execution stops gracefully, then the connection is closed.

D. It is not possible to close the connection while debugging is in progress.

Answer: C

Explanation:

When you close a remote debugging connection while a debugging execution is in progress, the execution is stopped gracefully on the remote robot and the connection is closed. This means that the

robot will finish the current activity and then stop the execution, without throwing any exception or error. This is the recommended way to end a remote debugging session, as it ensures that the robot is in a consistent state and ready for the next execution.

Reference: [Remote Debugging]

3.Where is the TransactionNumber incremented in the REFramework?

A. Only in the RetryCurrentTransaction.xaml workflow.

B. Only in the SetTransactionStatus.xaml workflow.

C. In the New Transaction transition.

D. In the RetryCurrentTransaction.xaml workflow and in the SetTransactionStatus.xaml workflow.

Answer: B

Explanation:

The TransactionNumber is incremented only in the SetTransactionStatus.xaml workflow, which is invoked at the end of each transaction. This workflow updates the status of the current transaction item in Orchestrator and increments the TransactionNumber by one. The TransactionNumber is used to keep track of the number of transactions processed by the robot and to get the next transaction item from the queue.

Reference: [SetTransactionStatus.xaml]

4.Which logging level includes the following information by default?

1. Execution Started log entry - generated every time a process is started.

2. Execution Ended log entry - generated every time a process is finalized.

3. Transaction Started log entry - generated every time a transaction item is obtained by the robot from Orchestrator.

4. Transaction Ended log entry - generated every time the robot sets the transaction status to either Success or Failed.

5. Activity Information log entry - generated every time an activity is started, faulted or finished inside a workflow.

6. Arguments and Variables Information log entry - show values of the variables and arguments that are used.

A. Verbose

B. Trace

C. Critical

D. Information

Answer: A

Explanation:

The Verbose logging level includes the following information by default:

Execution Started log entry - generated every time a process is started.

Execution Ended log entry - generated every time a process is finalized.

Transaction Started log entry - generated every time a transaction item is obtained by the robot from Orchestrator.

Transaction Ended log entry - generated every time the robot sets the transaction status to either Success or Failed.

Activity Information log entry - generated every time an activity is started, faulted or finished inside a

workflow.

Arguments and Variables Information log entry - show values of the variables and arguments that are used.

The Verbose logging level is the most detailed one and it is useful for debugging purposes, as it provides a lot of information about the execution flow and the data used by the robot. However, it also generates a lot of log entries, which can affect the performance and the storage space of the robot and Orchestrator.

Reference: [Logging Levels]

5.How would you define a linear process in UiPath?

A. The steps of the process refer to the execution of steps in a sequential manner, where each subsequent step depends on the successful completion of the previous step.

B. The steps of the process are performed multiple times, but each time different data items are used.

C. The steps of the process repeat multiple times over different data items. However, the automation design is such that each repeatable part processes independently.

D. The process steps are performed only once. If the need is to process additional data, then the automation must execute again.

Answer: D

Explanation:

A linear process in UiPath is a type of process that is executed only once and does not involve any looping or branching logic. It is suitable for simple scenarios where the input data is fixed and the output is predictable. A linear process can be designed using a Sequence or a Flowchart diagram, but it does not use any Flow Decision, Switch, While, Do While, or For Each activity. If the process needs to process additional data, then the automation must be executed again with the new data as input.

Reference: Framework for linear process or single transaction, How to modify ReFramework to Linear Process, Workflow Design, Difference between Linear process and Transactional process