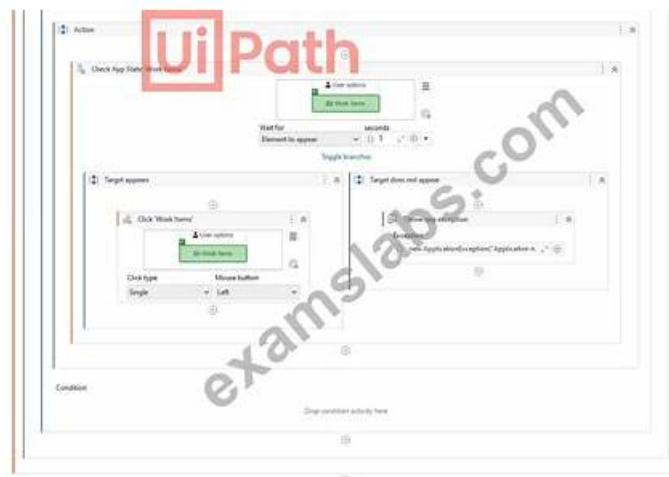


UiPath-ADAv1在線題庫， UiPath-ADAv1更新



我們會在互聯網上免費提供部分關於UiPath UiPath-ADAv1 認證考試的練習題讓嘗試，您會發現Fast2test的練習題是最全面的，是你最想要的。

UiPath UiPath-ADAv1 考試大綱：

主題	簡介
主題 1	<ul style="list-style-type: none">UI Automation: Here, the topic explains how UI Automation works and offer guidance on using the Modern Recorder and associated activities. It also covers UI synchronization and the configuration of static and dynamic descriptors.
主題 2	<ul style="list-style-type: none">Business Knowledge: This topic covers the fundamental concepts of business process automation, highlighting its value proposition. It also explores key ideas related to business processes, offering a comprehensive understanding of this domain.
主題 3	<ul style="list-style-type: none">Integration Service: The section introduces Integration Service, explaining its purpose and demonstrating the use of connectors and triggers in automation projects to interact with external systems.
主題 4	<ul style="list-style-type: none">Libraries and Templates: This topic covers the creation, publication, and consumption of process libraries, along with the sharing and access of templates, promoting efficient project development and standardization.
主題 5	<ul style="list-style-type: none">Exception Handling: This topic focuses on error management, showcasing the use of Try Catch, Throw, and Rethrow activities, along with the Retry Scope feature, to handle exceptions gracefully.
主題 6	<ul style="list-style-type: none">Implementation Methodology: The section offers an overview of project implementation stages, interpretation of PDDs and SDDs, and the conduct of automation project peer reviews, ensuring a structured approach to development.
主題 7	<ul style="list-style-type: none">Email Automation: This topic covers retrieving emails via IMAPPOP3, sending SMTP messages, and managing integrations with Microsoft and Gmail accounts, utilizing their respective packages.
主題 8	<ul style="list-style-type: none">Workflow Analyzer: Here, the topic introduces the Workflow Analyzer tool, explaining its use for project analysis and validation, and providing guidance on configuring its settings.
主題 9	<ul style="list-style-type: none">Object Repository: This topic covers the creation, publication, and consumption of UI Libraries, including the use of static and dynamic descriptors, offering a structured approach to UI element management.

主題 10	<ul style="list-style-type: none"> PDF Automation: The section focuses on data extraction from native and scanned PDFs, including single and multiple document scenarios.
主題 11	<ul style="list-style-type: none"> Debugging: Here, we explore various debugging techniques, including debug modes, actions, and ribbon options. It also guides users through setting breakpoints, utilizing debugging panels, and optimizing performance with profile execution.
主題 12	<ul style="list-style-type: none"> Platform Knowledge: The section provides an overview of UiPath's product suite, including Studio and Robot variants, Orchestrator, and Integration Service, showcasing their unique contributions. It also emphasizes the benefits of Academy, Forum, and Marketplace in the UiPath ecosystem.
主題 13	<ul style="list-style-type: none"> Version Control Integration: The section highlights the benefits of version control by demonstrating the use of Studio's Git integration for adding projects, cloning repositories, committing changes, and managing branches.
主題 14	<ul style="list-style-type: none"> Orchestrator: This topic covers the definition of Orchestrator entities, tenant entities, and folder entities, along with their respective functionalities. It also provides practical guidance on robot provisioning, workspace management, role assignments, and logging features.
主題 15	<ul style="list-style-type: none"> Excel Automation: The section delves into Excel Automation, showcasing the use of modern Excel activities and workbook-level operations.
主題 16	<ul style="list-style-type: none"> Logging: The section provides insights into interpreting robot execution logs and adhering to logging best practices, ensuring effective issue identification and resolution.
主題 17	<ul style="list-style-type: none"> Studio Interface: Here, the topic guides users through installing Studio Community Edition and connecting to Orchestrator. It covers profile differences, backstage view options, compatibility modes, and package management. Additionally, it offers an in-depth exploration of the Studio interface and its various elements.
主題 18	<ul style="list-style-type: none"> Control Flow: The section explains the functionality of control flow activities and their impact on workflow types. It covers the use of sequence and flowchart layouts. Lastly, it focuses on implementing IF, For Each, and While activities, among others.
主題 19	<ul style="list-style-type: none"> Variables and Arguments: This topic introduces data types and delves into creating, managing, and utilizing variables, arguments, and global constants variables. It also clarifies the distinctions between these concepts, ensuring a comprehensive understanding of data handling in automation projects.

>> UiPath-ADAv1在線題庫 <<

UiPath-ADAv1更新 - UiPath-ADAv1題庫分享

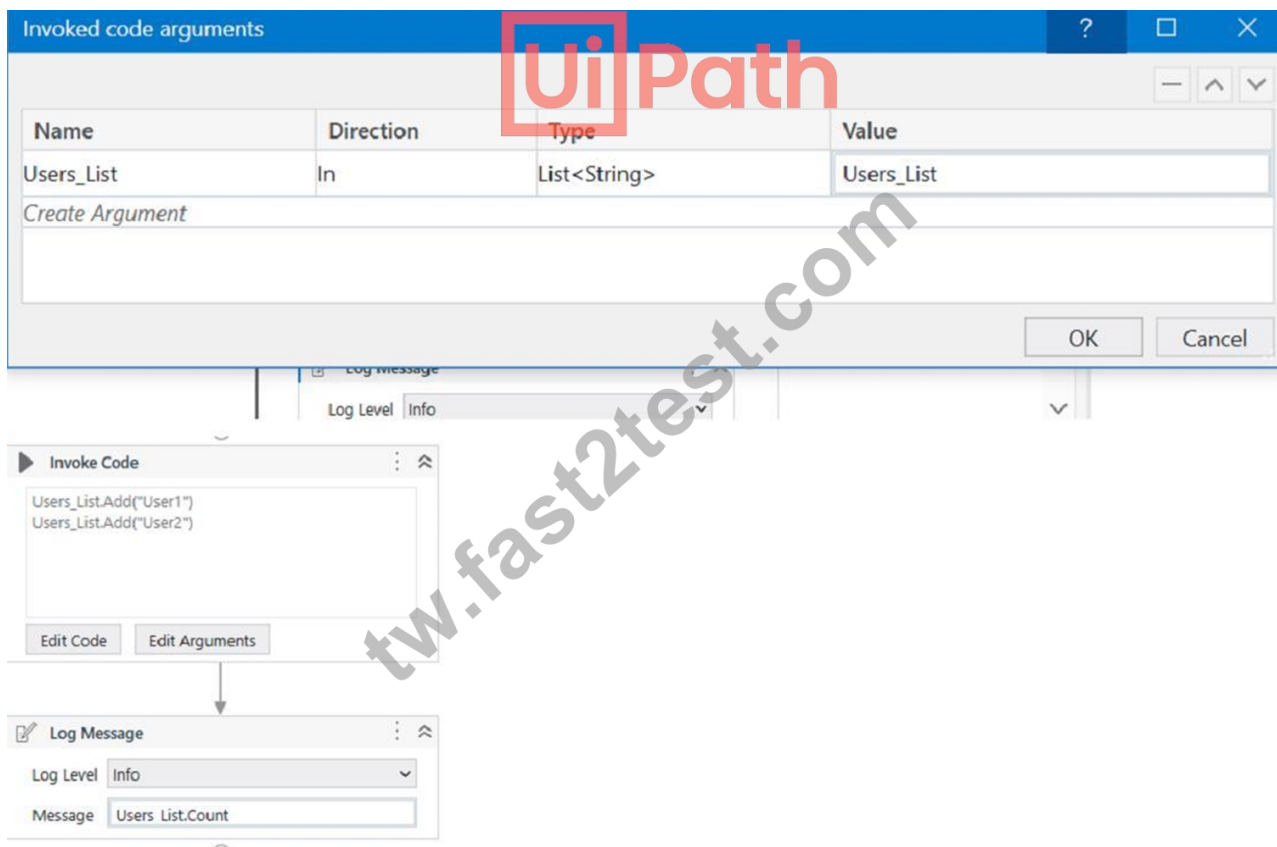
UiPath UiPath-ADAv1認證考試是目前IT人士報名參加的考試中很受歡迎的一個認證考試。通過了UiPath UiPath-ADAv1認證考試不僅能使你工作和生活帶來提升，而且還能鞏固你在IT領域的地位。但是事實情況是它通過率確很低。

最新的 UiPath Certified Professional - Developer Track UiPath-ADAv1 免費考試真題 (Q144-Q149):

問題 #144

A developer has created a variable of type List of Strings named "UsersList", and initialized it with an empty list: "UsersList = new List(Of String)".

What is printed in the log message after the following Invoke Code is executed?



- A. 0
- B. Object reference not set to an instance exception is thrown
- C. System.ArgumentException is thrown
- D. 1

答案: A

解題說明:

Explanation

The code is adding two items to the list, but since the list is initialized as an empty list, the count will be 0.

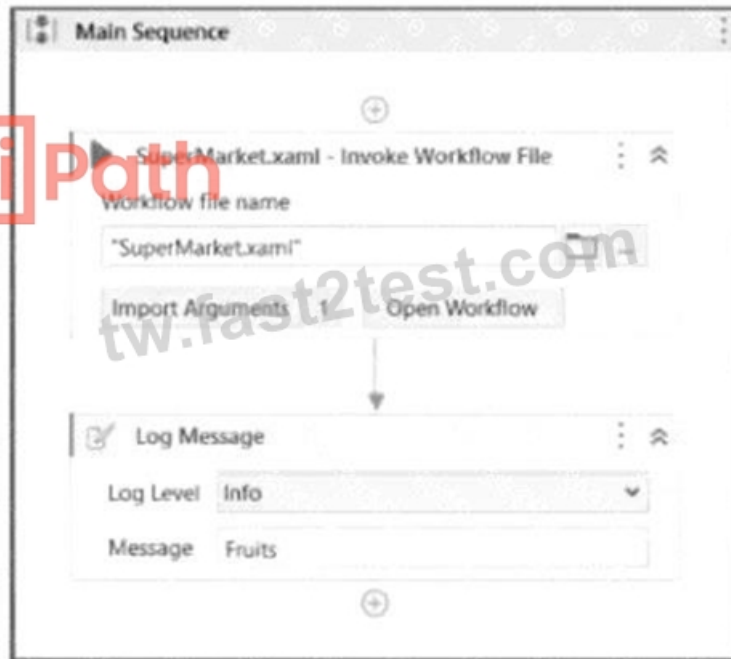
References:

Data Manipulation with Lists and Dictionaries in Studio course, Lesson 2: Lists, Topic: Creating Lists How to initiate List<String[]>? forum post, Answer by @VishalS

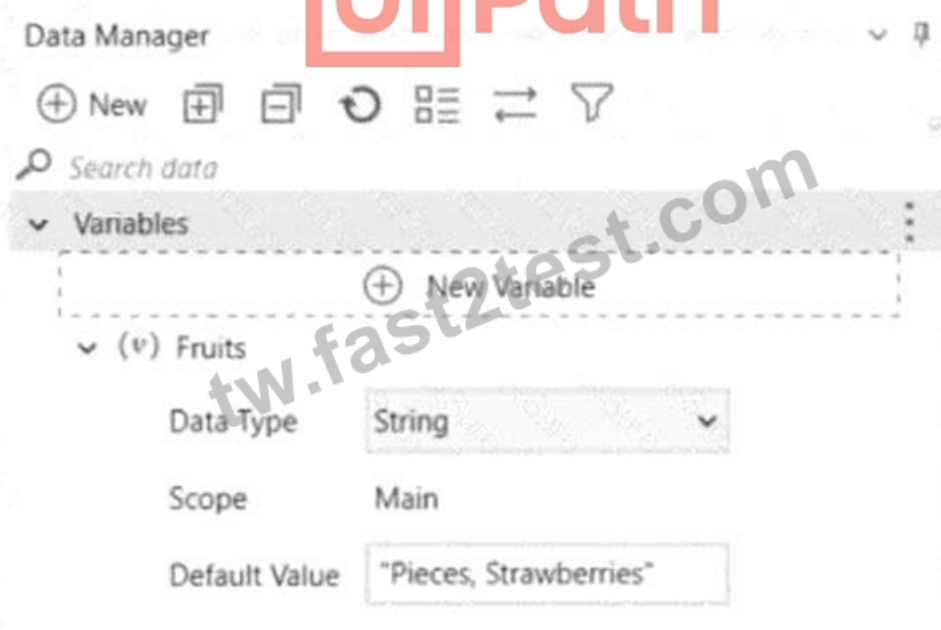
問題 #145

A developer has designed an automation workflow which comprises:

1. Main.xaml file:



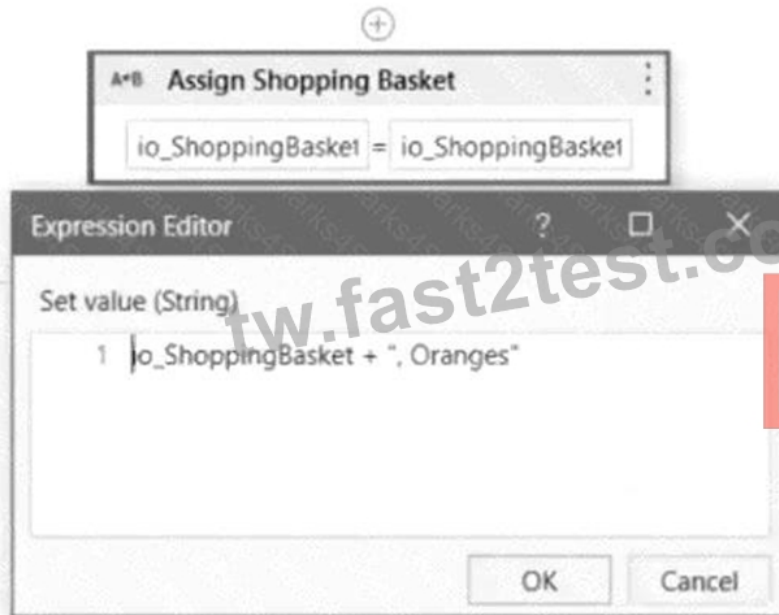
2. Variables panel in the Main.xaml file:



3. Main.xaml file invokes the SuperMarket workflow using the following argument:



4. Detailed SuperMarket sequence and right-hand side expression shown below:



Based on the information shown in the exhibits, what is the output of the Write Line activity in the "Main.xaml" file?

- A. Pieces, Strawberries, Oranges
- B. Oranges
- C. , Oranges
- D. Pieces, Strawberries

答案：A

解題說明：

Based on the information from the exhibits, the "Main.xaml" file has a variable Fruits with a default value "Pieces, Strawberries". This variable is passed as an in/out argument `io_ShoppingBasket` to the "SuperMarket.xaml" workflow. Inside the "SuperMarket.xaml" workflow, "Oranges" is appended to `io_ShoppingBasket`. Since `io_ShoppingBasket` is an in/out argument, the value of Fruits in the "Main.xaml" file will be updated to "Pieces, Strawberries, Oranges". Therefore, the output of the Write Line activity in the "Main.xaml" file will be "Pieces, Strawberries, Oranges".

問題 #146

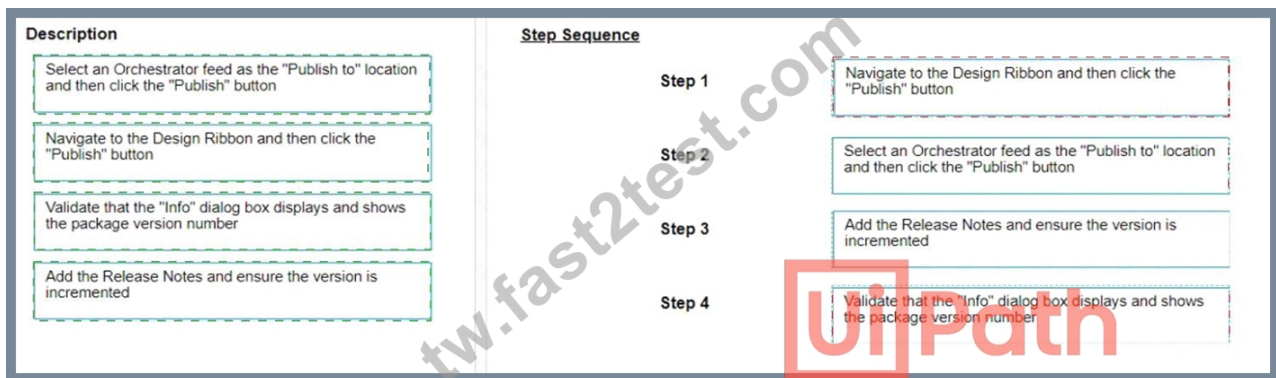
A developer finished creating an automation project in UiPath Studio 2021.10. What is the recommended sequence of steps that should be performed to publish the package to Orchestrator 2021.10?

Instructions: Drag the Description found on the left and drop on the correct Step found on the right

Description	Step Sequence
Select an Orchestrator feed as the "Publish to" location and then click the "Publish" button	Step 1
Navigate to the Design Ribbon and then click the "Publish" button	Step 2
Validate that the "Info" dialog box displays and shows the package version number	Step 3
Add the Release Notes and ensure the version is incremented	Step 4

答案：

解題說明：



Explanation

The recommended sequence of steps that should be performed to publish the package to Orchestrator 2021.10 is:

Step 1 = Navigate to the Design Ribbon and then click the "Publish" button
 Step 2 = Select an Orchestrator feed as the "Publish to" location and then click the "Publish" button
 Step 3 = Add the Release Notes and ensure the version is incremented
 Step 4 = Validate that the "Info" dialog box displays and shows the package version number
 This sequence will ensure that the developer can publish the automation project to the Orchestrator feed of their choice, add some notes about the changes made, and verify that the package is successfully published with the correct version number.

<https://docs.uipath.com/orchestrator/standalone/2021.10/user-guide/about-packages>

問題 #147

To retrieve all Outlook emails with the word "UiPath" in the subject, which filtering property and filter schema should a developer use?

- A. Property: FilterByMessageIds
Schema: "@SQL="urn:schemas:httpmail:subject" like 'UiPath%'"
- B. Property: FilterByMessageIds
Schema: "@SQL="urn:schemas:httpmail:subject" like '%UiPath%'"
- C. Property: Filter
Schema: "@SQL="urn:schemas:httpmail:subject" like 'UiPath%'"
- D. Property: Filter
Schema: "@SQL="urn:schemas:httpmail:subject" like '%UiPath%'"

答案: D

解題說明:

To retrieve all Outlook emails with the word "UiPath" in the subject, a developer should use the Filter property and the filter schema "@SQL="urn:schemas:httpmail:subject" like '%UiPath%'" . The Filter property allows developers to specify a DASL query that filters the emails based on various criteria, such as subject, sender, date, etc. The filter schema

"@SQL="urn:schemas:httpmail:subject" like '%UiPath%'" uses the SQL dialect of DASL to search for emails that have the word "UiPath" anywhere in the subject. The

"%" symbol is a wildcard that matches any character or string. (UiPath Automation Developer study guide)

問題 #148

A variable should be created and used for storing one or multiple IDs. The collection of IDs can be updated during the execution of the robot by adding elements. Which should be the data type of this variable, taking into account that each of the IDs has the following format: XX-XX-XX (where X is a digit)?

- A. String[]
- B. String
- C. List<Int32>
- D. List<String>

答案: D

解題說明:

For storing one or multiple IDs with the format XX-XX-XX (where X is a digit) and allowing for updates during execution, the appropriate data type is List<String>. This is because you need a collection that can dynamically expand as more IDs are added,

問題 #149

UiPath-ADAv1更新: <https://tw.fast2test.com/UiPath-ADAv1-premium-file.html>

- [illegible]