

試験の準備方法-高品質なUiPath-ADAv1認定試験トレーニング試験-検証するUiPath-ADAv1基礎訓練



BONUS!!! GoShiken UiPath-ADAv1ダンプの一部を無料でダウンロード: https://drive.google.com/open?id=1kju-32_INnJn-i_Elj35EDP7sB5AUMc

テストに関する最も有用で効率的なUiPath-ADAv1トレーニング資料を提供するために最善を尽くし、クライアントが効率的に学習できるように複数の機能と直感的な方法を提供します。UiPath-ADAv1の有用なテストガイドを学習すれば、時間と労力はほとんどかかりません。合格率とヒット率はともに高いため、テストに合格するための障害はほとんどありません。Webで紹介を読んだ後、UiPath-ADAv1学習実践ガイドをさらに理解できます。

UiPath UiPath-ADAv1 認定試験の出題範囲:

トピック	出題範囲
トピック 1	<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.
トピック 2	<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.
トピック 3	<ul style="list-style-type: none">Working with Files and Folders: Here, you explore creating, managing, and navigating local files and folders, providing a foundation for file system automation.
トピック 4	<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.
トピック 5	<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.
トピック 6	<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.
トピック 7	<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.

トピック 8	<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.
トピック 9	<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.
トピック 10	<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.
トピック 11	<ul style="list-style-type: none"> Data Manipulation: This topic covers a range of data manipulation techniques, including string manipulation using VB.Net methods and the RegEx Builder. It also guides users through working with arrays, lists, and dictionaries, as well as building and iterating through DataTables.
トピック 12	<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.
トピック 13	<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.
トピック 14	<ul style="list-style-type: none"> Email Automation: This topic covers retrieving emails via IMAP POP3, sending SMTP messages, and managing integrations with Microsoft and Gmail accounts, utilizing their respective packages.

>> UiPath-ADAv1認定試験トレーニング <<

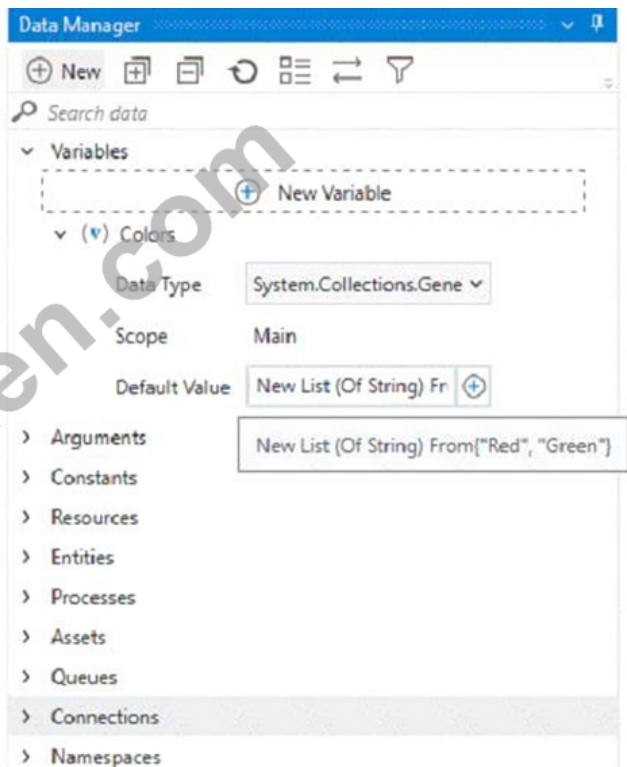
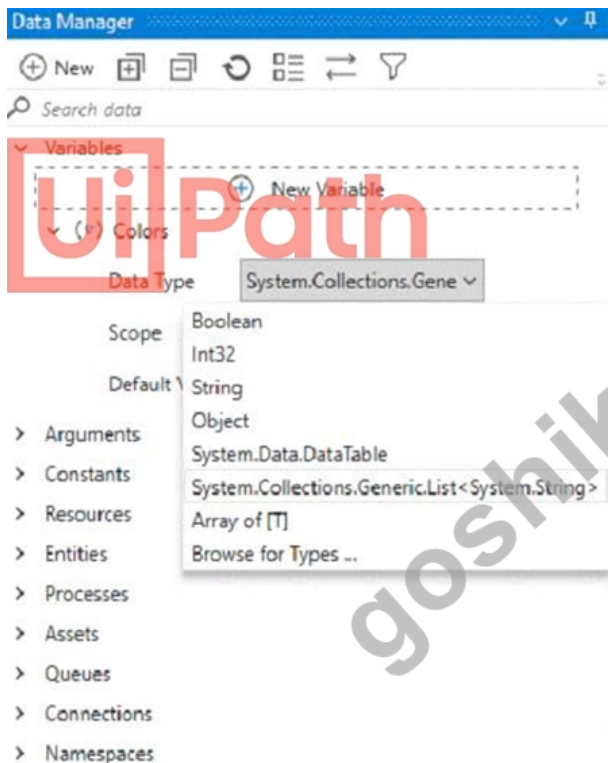
UiPath-ADAv1基礎訓練、UiPath-ADAv1問題無料

多くの人々は、ある分野での仕事に秀でることができ、知識をある産業での実際の仕事に応用するのに熟練した有能な人になりたいと思っています。しかし、彼らにとっては簡単なことではなく、目標を達成するために多くの努力が必要です。テストUiPath-ADAv1認定に合格すると、彼らはそのような人々になります。あなたが彼らの1人であれば、UiPath-ADAv1学習教材を購入することで、少ない労力でスムーズにテストに合格できます。UiPath-ADAv1試験の質問は価値があり、有用です。当社の製品を購入すると、最高のサービスを提供して満足することができます。

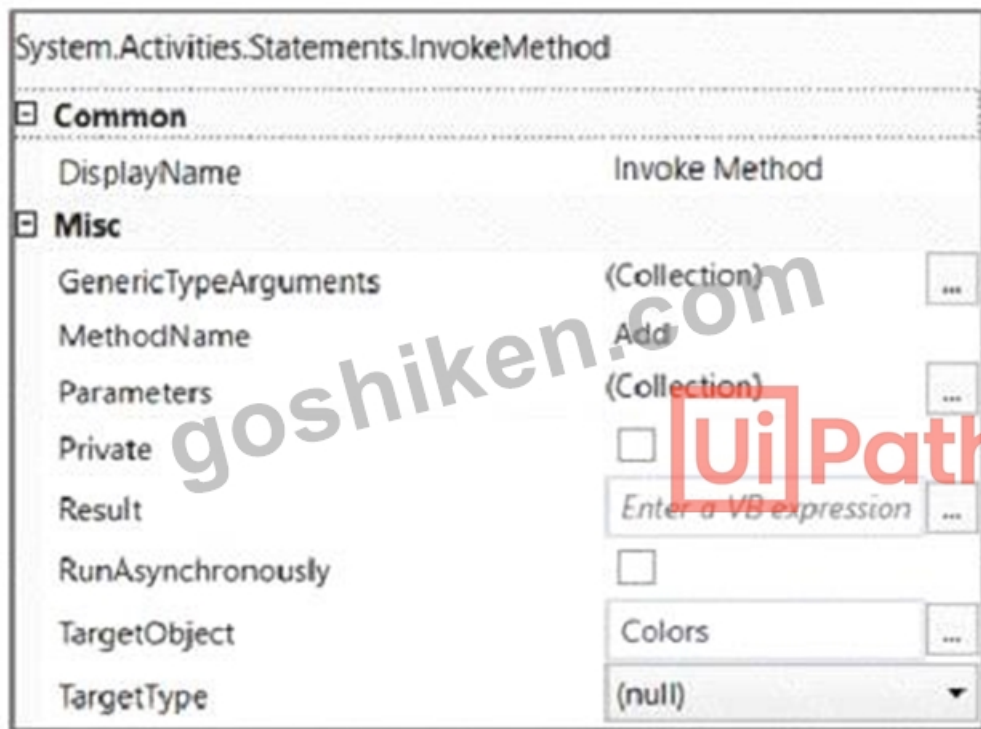
UiPath Automation Developer Associate v1 Exam 認定 UiPath-ADAv1 試験問題 (Q320-Q325):

質問 # 320

A developer wants to add items to a list of strings using the Invoke Method activity. The list is declared as follows:



The Invoke Method includes the following properties:



The Parameters property is as follows:



Based on the exhibits, what is the outcome of this Invoke Method activity?

- A. Colors will contain items in the following order: "Yellow", "Red", "Green".
- **B. Colors will contain items in the following order: "Red", "Green", "Yellow".**
- C. Colors will contain items in the following order: "Red", "Green".
- D. Invoke Method activity will throw an error.

正解: B

解説:

Explanation

The Invoke Method activity is used to execute a method of a class or an object¹. In this case, the developer wants to add items to a list of strings using the Add method of the List class². The list is declared as Colors and initialized with two items: "Red" and "Green". The Invoke Method activity has the following properties:

TargetObject: Colors (the list variable)

MethodName: Add (the method of the List class)

Parameters: Direction - In, Type - String, Value - "Yellow" (the item to be added to the list) Based on these properties, the Invoke Method activity will add the string "Yellow" to the end of the Colors list. Therefore, the outcome of this Invoke Method activity is that Colors will contain items in the following order: "Red", "Green", "Yellow".

Option A is incorrect because the Invoke Method activity will not throw an error, as the properties are configured correctly. Option

B is incorrect because the order of the items in the list will not change, as the Add method appends the item to the end of the list.

Option D is incorrect because the list will have three items, not two, as the Add method does not overwrite any existing item.

References:

Invoke Method activity documentation from UiPath

List<T>.Add(T) Method documentation from Microsoft

質問 # 321

A developer stores value "25.11" in a String variable called "InvoiceTotal". Which expression should be used to convert "InvoiceTotal" to a numeric format with the decimals included?

- A. CInt(Invoice Total)
- **B. Double.Parse(InvoiceTotal)**
- C. String.ToNumeric(Invoice Total)
- D. Invoice Total.IsNumeric

正解: B

解説:

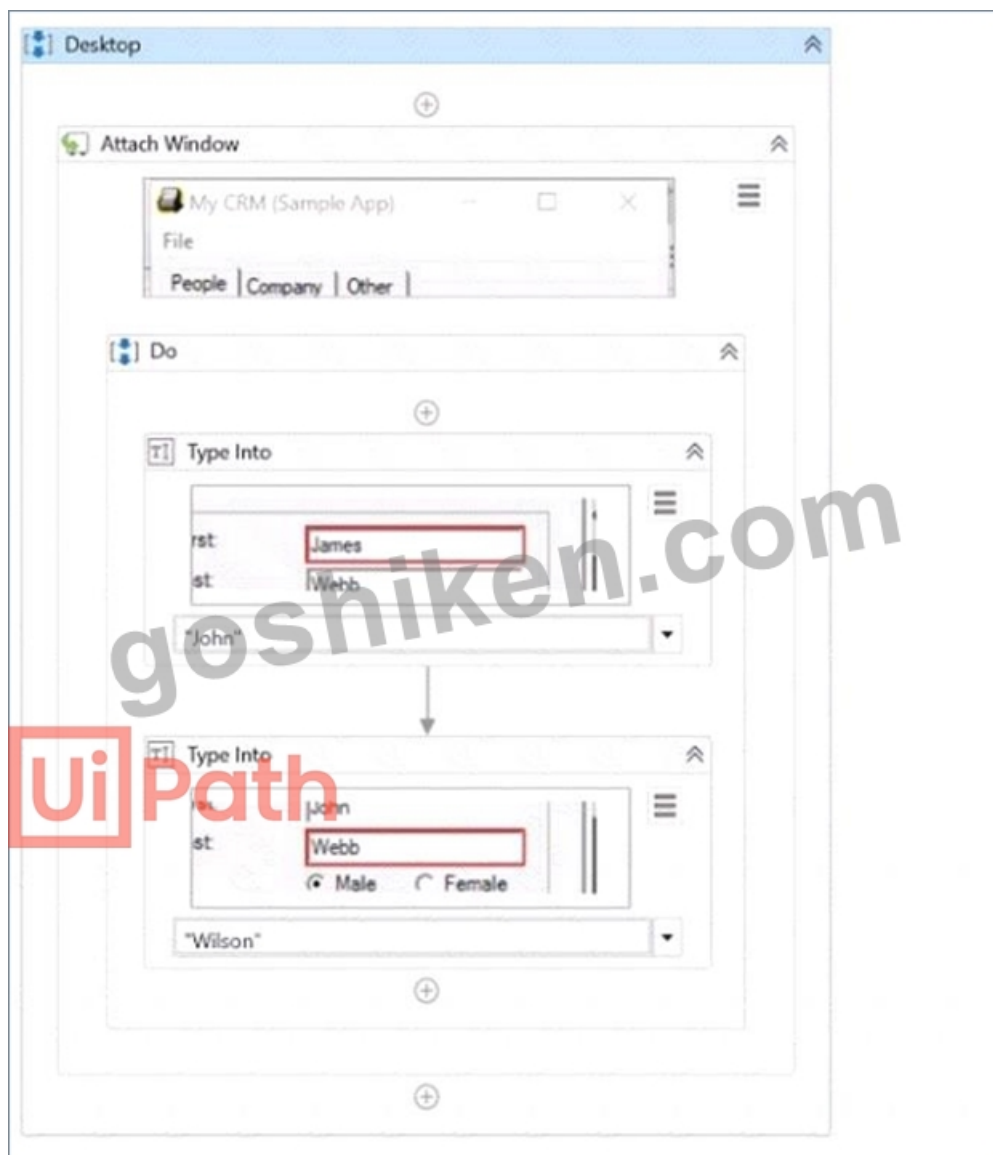
To convert a string variable "InvoiceTotal" to a numeric format that includes decimals, the expression Double.Parse(InvoiceTotal) should be used. This method converts the string representation of a number in a specified style and culture-specific format to its double-precision floating-point number equivalent.

References:

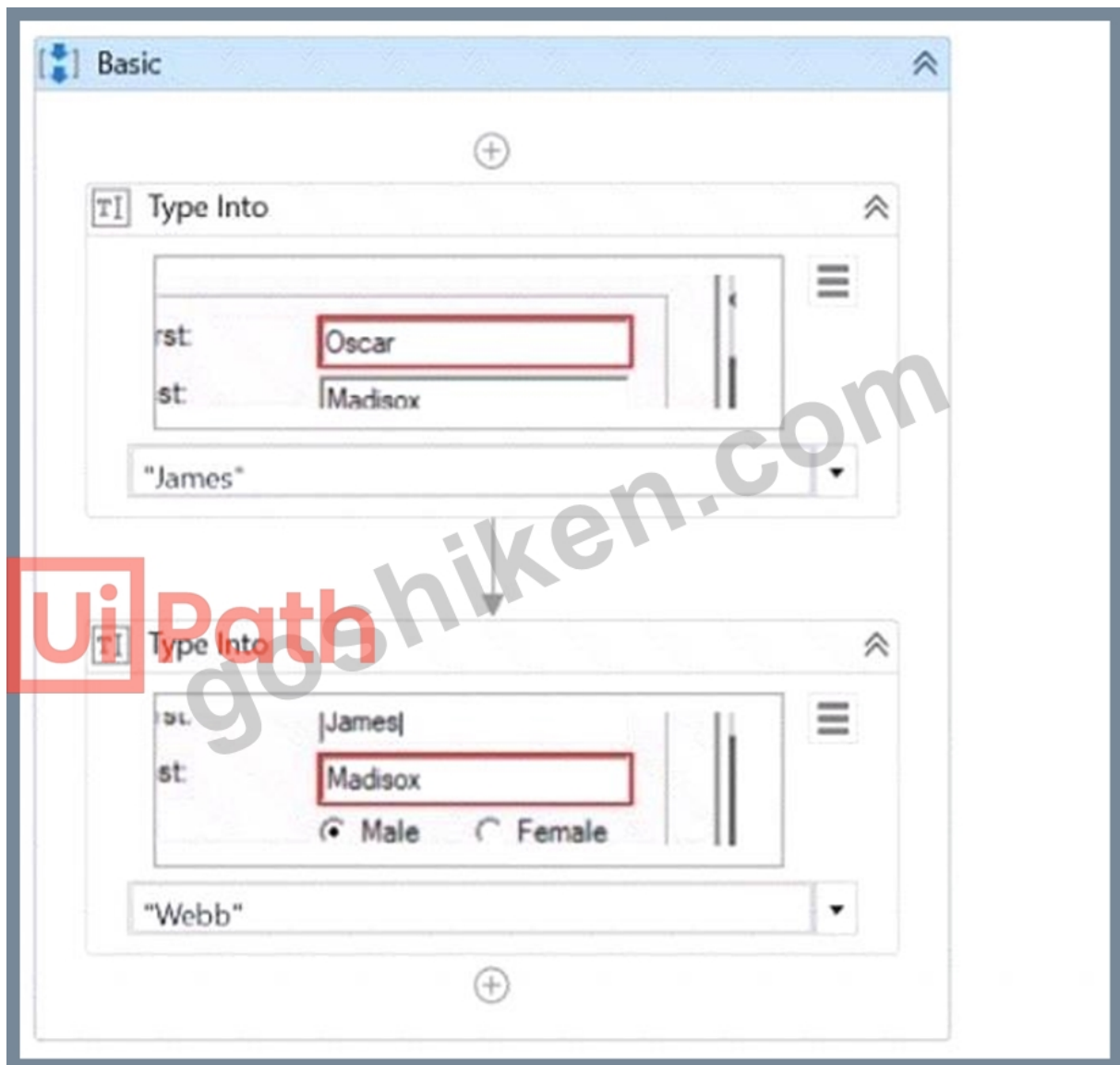
Microsoft .NET Documentation: Double.Parse Method

質問 # 322

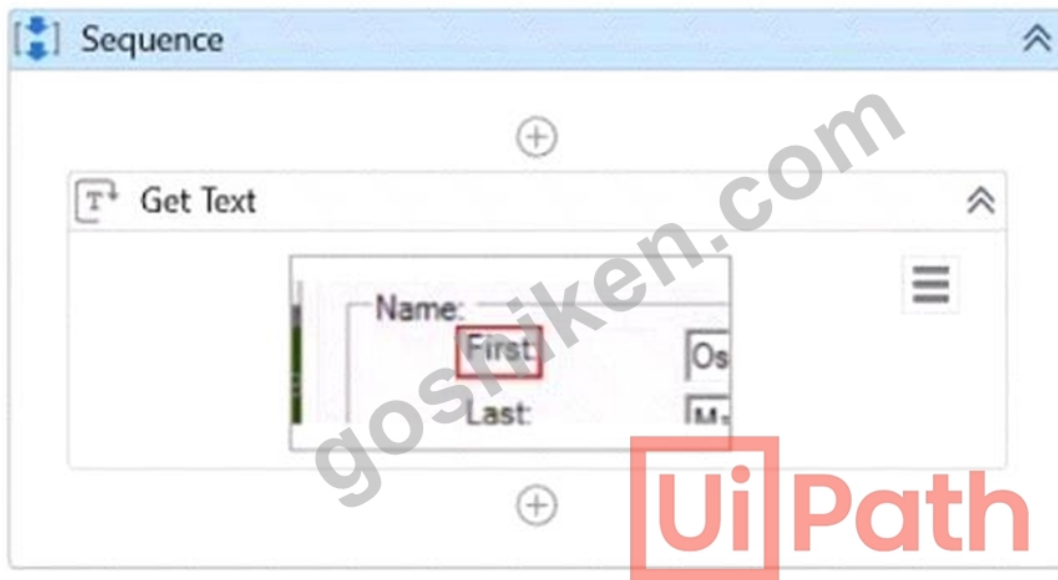
What represents a sequence that contains an activity with a partial selector?



- A.
- B.



- C.
- D.



正解: C

解説:

A partial selector is a selector that does not contain the information about the top-level window. A partial selector is usually generated when a UI Automation activity is included in a container activity, such as Attach Browser or Attach Window, that has a full selector of the outer window. This way, the UI Automation activity can use the partial selector relative to the container, instead of the full selector from the root. A partial selector is indicated by the icon in the Properties panel of the UI Automation activity. In this question, the sequence that contains an activity with a partial selector is Option B, where the Click activity has the partial selector icon and is enclosed in an Attach Window activity. The other options have activities with full selectors, indicated by the icon. Therefore, the answer is B. Option B. References: Full Versus Partial Selectors, Selectors

質問 # 323

The provided exhibit displays the representation of the "Deadline" field using the yyyy-mm-dd format.

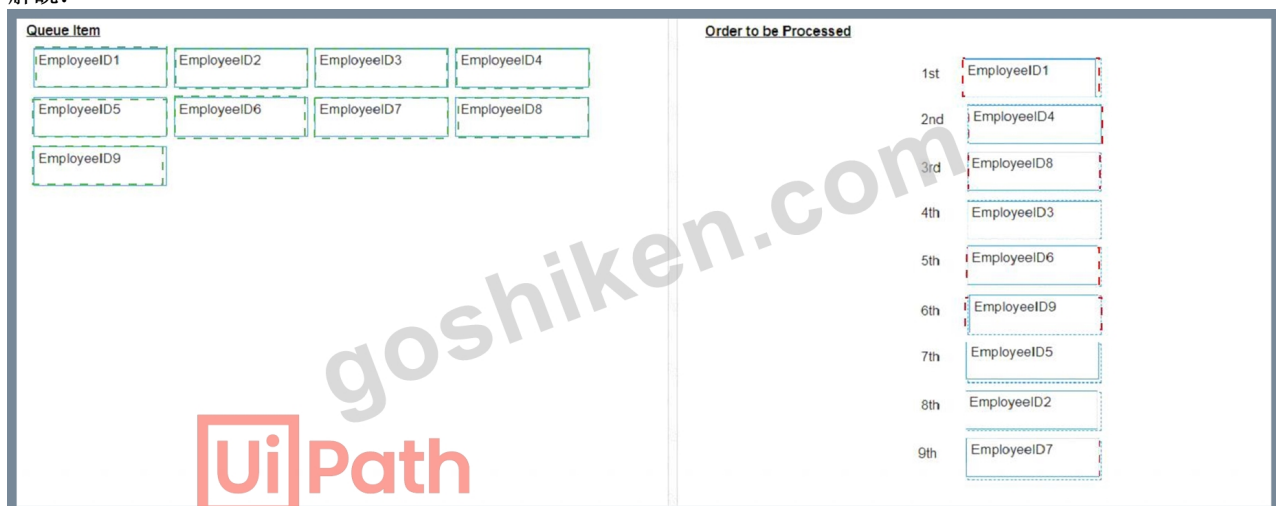
Queue Item Name	Deadline	Priority
EmployeeID1		Normal
EmployeeID2	2050-01-01	High
EmployeeID3	2050-01-02	High
EmployeeID4	2050-01-01	Normal
EmployeeID5	2050-01-02	Low
EmployeeID6		High
EmployeeID7	2050-01-01	Low
EmployeeID8	2050-01-01	Normal
EmployeeID9		Low

In what order should the Queue Items be processed by a single robot executing a Performer workflow?



正解:

解説:



Explanation:

The Queue Items should be processed by a single robot executing a Performer workflow in the following order:

EmployeeID1
EmployeeID4
EmployeeID8
EmployeeID3
EmployeeID6
EmployeeID9
EmployeeID5
EmployeeID2
EmployeeID7

A queue is a container that holds an unlimited number of items that can be processed by a robot¹. A queue item can have a deadline and a priority, which affect the order in which the robot retrieves the items from the queue². The Performer workflow is a part of the REFramework template that processes the queue items using the Get Transaction Item activity³. The order of queue execution is determined by the following rules⁴:

Items that have a deadline are processed first, in order of priority and according to the set deadline for items with the same priority. Items that have no deadline are processed next, in order of priority and according to the rule First In, First Out for items with the same priority.

Based on these rules and the exhibit, the order of queue execution is as follows:

EmployeeID1 has the earliest deadline (2050-01-01) and high priority, so it is processed first.

EmployeeID4 has the same deadline as EmployeeID1, but normal priority, so it is processed second.

EmployeeID8 has the same deadline and priority as EmployeeID4, but it was added later to the queue, so it is processed third.

EmployeeID3 has the next earliest deadline (2050-01-02) and high priority, so it is processed fourth.

EmployeeID6 has the same deadline as EmployeeID3, but normal priority, so it is processed fifth.

EmployeeID9 has the same deadline and priority as EmployeeID6, but it was added later to the queue, so it is processed sixth.

EmployeeID5 has no deadline, but high priority, so it is processed seventh.

EmployeeID2 has no deadline and normal priority, so it is processed eighth.

EmployeeID7 has no deadline and low priority, so it is processed ninth.

References:

About Queues and Transactions documentation from UiPath

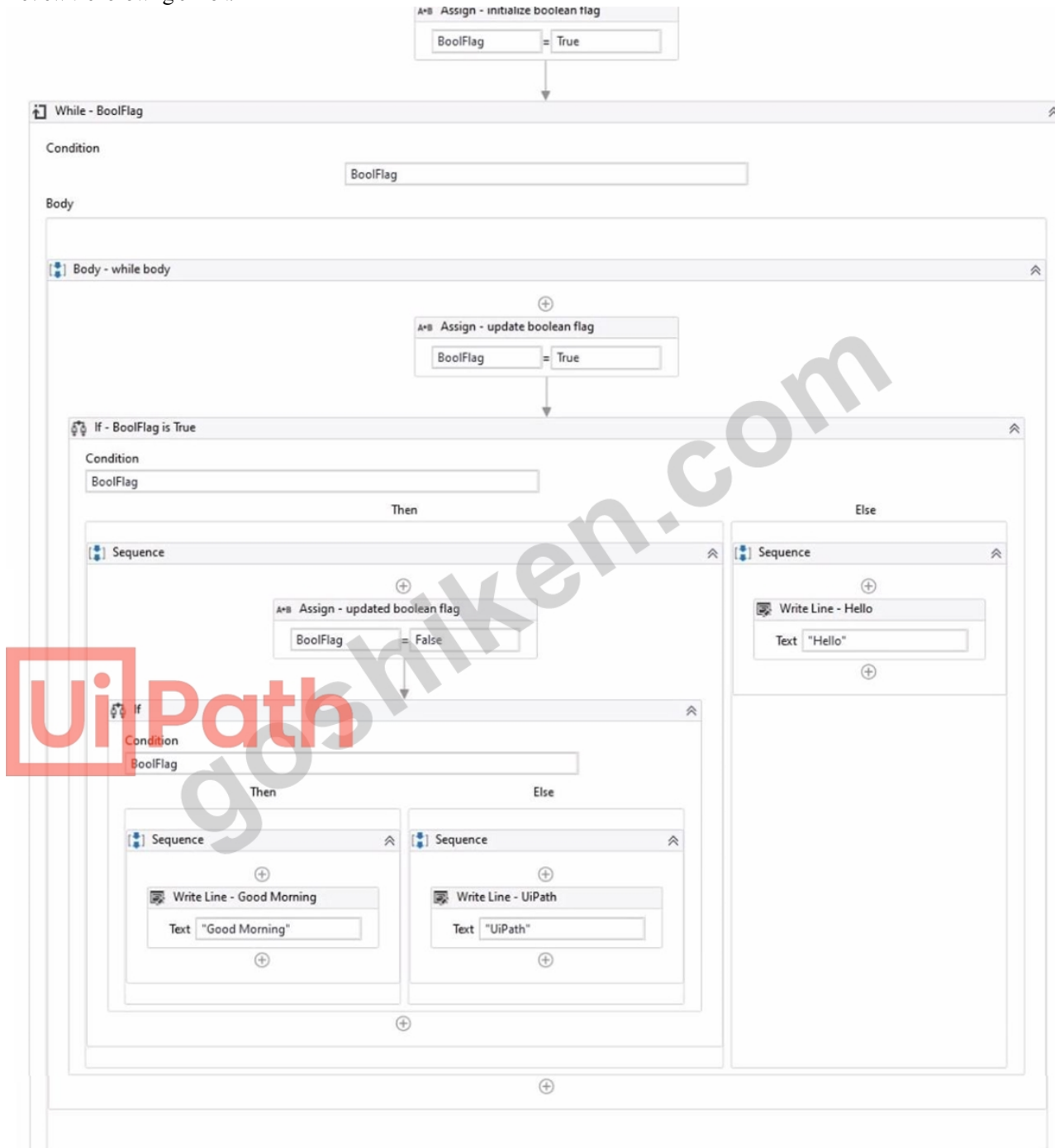
Processing Order documentation from UiPath

Performer documentation from UiPath

Order processing queue forum post from UiPath Community

質問 # 324

Review the following exhibit.



Based on the exhibit, which output is produced in the Output panel?

- A. Hello
- B. Good Morning UiPath
- C. Good Morning
- D. UiPath

解説:

Based on the exhibit, the output produced in the Output panel is "Good Morning". This is because the "If" activity checks if the "BoolFlag" variable is True. If it is True, the "Then" branch is executed, which contains a "Write Line" activity with the text "Good Morning". If the "BoolFlag" variable is False, the "Else" branch is executed, which contains a "Write Line" activity with the text "UiPath". Since the "BoolFlag" variable is assigned to True in the previous "Assign" activity, the condition of the "If" activity is satisfied, and the "Then" branch is executed. Therefore, the "Write Line" activity writes "Good Morning" to the Output panel⁴.
References: If and How to put a if condition in assign activity? from UiPath documentation and forum.

• • • • •

UiPath-ADAv1基礎訓練: <https://www.goshiken.com/UiPath/UiPath-ADAv1-mondaishu.html>

- BONUS!!! GoShiken UiPath-ADAv1ダンプの一部を無料でダウンロード: <https://drive.google.com/open?id=1kiv->

