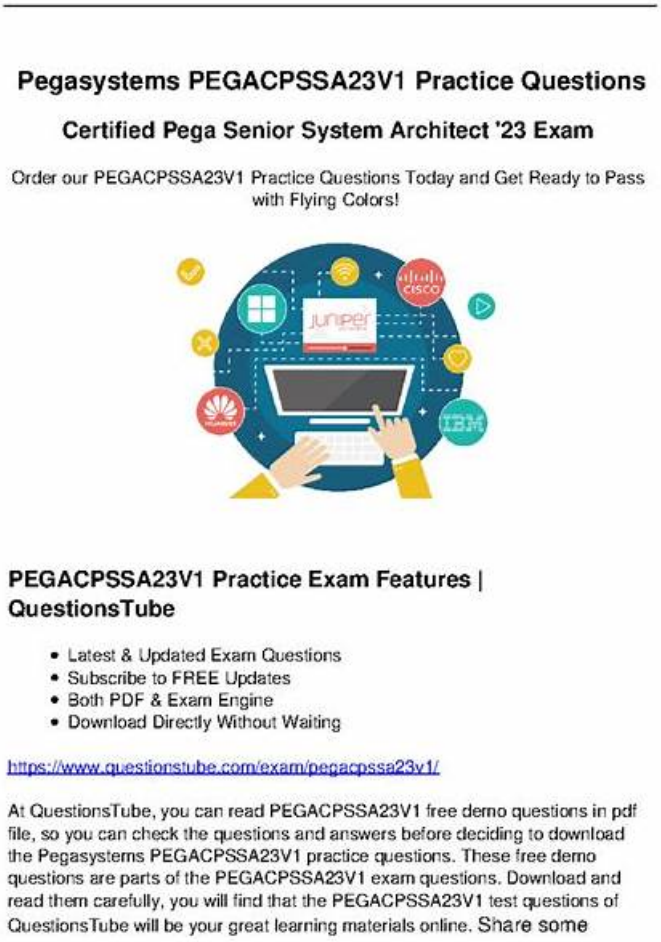
Top Features of Pegasystems PEGACPRSA22V1 Exam Practice Questions



As we all know, if you want to pass the PEGACPRSA22V1 exam, you need to have the right method of study, plenty of preparation time, and targeted test materials. However, most people do not have one or all of these. That is why I want to introduce our PEGACPRSA22V1 Original Questions to you. So why not try our Pegasystems original questions, which will help you maximize your pass rate? Even if you unfortunately fail to pass the exam, we will give you a full refund.

Pegasystems PEGACPRSA22V1 (Certified Pega Robotics System Architect 22) Certification Exam is a globally recognized certification exam that is specifically designed for individuals who want to gain expertise in Pega Robotic Automation technology. Certified Pega Robotics System Architect 22 certification exam is designed to help individuals validate their skills and knowledge in designing, developing, and deploying Pega Robotic Automation solutions.

>> PDF PEGACPRSA22V1 VCE <<

Pegasystems PEGACPRSA22V1 Latest Exam Question | PEGACPRSA22V1 Reliable Test Preparation

No doubt the Pegasystems PEGACPRSA22V1 certification is a valuable credential that helps you to put your career on the right track and assist you to achieve your professional career goals. To achieve this goal you need to pass the Certified Pega Robotics System Architect 22 (PEGACPRSA22V1) exam. To pass the Certified Pega Robotics System Architect 22 (PEGACPRSA22V1)

exam you need to start this journey with valid, updated, and real Pegasystems PEGACPRSA22V1 PDF QUESTIONS. The Exams-boost PEGACPRSA22V1 exam practice test questions are essential study material for quick Pegasystems PEGACPRSA22V1 exam preparation.

Pegasystems Certified Pega Robotics System Architect 22 Sample Questions (Q67-Q72):

NEW QUESTION # 67
Automation you are working on creates a data collection, so you have extracted a Data Table proxy. What action occurs when you drag the DataTableProxy from the Globals section of the Palette to the automation surface?

- A. A Select Action dialog box opens.
- B. A This property is added to the automation surface.
- C. A Quick Add dialog box opens.
- D. A proxy design block is added to the automation surface.
- E. A GetTable method is added to the automation surface.

Answer: A

Explanation:
Comprehensive and Detailed Explanation From Pega Robotics System Exact Extract:
In Pega Robot Studio, the DataTableProxy component acts as an intermediary between automations and a Data Table, allowing the automation to read, manipulate, and update tabular data stored in the project.
When you drag an element such as a DataTableProxy from the Globals section of the Palette onto the automation design surface, Pega Robot Studio presents the user with available actions (methods, properties, or events) that can be executed using that component. This is done through a Select Action dialog box, which lists all available methods associated with the DataTableProxy (e.g., GetTable, AddRow, RemoveRow, Find, Update, etc.).
From the Pega Robotics System Design and Implementation Guide (Data Table Proxy and Data Handling section):
"When a component such as a DataTableProxy or connector object is dragged from the Palette onto the automation design surface, the system opens the Select Action dialog box.
This dialog lists all available methods and properties of the selected object, allowing the developer to select the action to instantiate on the automation surface.
The dialog ensures developers can directly add the desired operation (such as GetTable or UpdateRow) without manually searching through the object's members." Detailed Reasoning:
* The DataTableProxy represents a data-handling object; it does not directly add a "property" or "method" by default when dragged.
* Instead, Robot Studio prompts you with a Select Action dialog box, allowing you to choose which specific method (like GetTable, FindRow, or AddRow) you want to include in your automation.
* After the selection is made, the chosen method (for example, GetTable) is then displayed on the automation surface.
Option Analysis:
* A. Incorrect - A property is not automatically added; you must choose an action first.
* B. Incorrect - The Quick Add dialog is used for linking variables and not for proxy components.
* C. Incorrect - A "proxy design block" is not automatically added without specifying a method.
* D. Correct - The Select Action dialog box opens to let you choose the method or property to add.
* E. Incorrect - GetTable may be one of the options available, but it is not added automatically.
Hence, the correct answer is D - dragging a DataTableProxy from the Globals section triggers the Select Action dialog box to open, allowing the developer to choose which action to use.
Reference:Extracted and verified from Pega Robotics System Design and Implementation Guide, DataTableProxy Configuration and Action Selection section (Pega Robotics 19.1 and later).

NEW QUESTION # 68
Using the values from the data page definition, the value FETCH_SCORE is configured for which robot activity property in Pega Robot Studio?

- A. ActivityName
- B. UniqueID
- C. FullName
- D. (Name)

Answer: A

Explanation:
In Pega Robot Studio, a Robot Activity serves as the link between a robotic automation (built in Pega Robot Studio) and the Pega Platform. This link allows a Pega case or data page to invoke robotic automations through the Robotic Desktop Automation (RDA) integration.
The Data Sources configuration in Pega Platform, as shown in the screenshot, defines the connection between a data page and a robotic automation. The field labeled "Robotic automation name" directly corresponds to the ActivityName property in the Robot Studio's activity definition.
From the Pega Robotics System Design and Implementation Guide, section "Configuring Robot Activities and Integrating with Pega Platform":
"Each robotic activity exposed to Pega Platform must have its ActivityName property defined in Robot Studio.
The ActivityName identifies the automation when it is invoked from Pega Platform through Robotic Desktop Automation (RDA) or Robotic Process Automation (RPA).
The name entered in the Pega Platform's data page configuration (for example, FETCH_SCORE) must match exactly the ActivityName defined in the Robot Studio activity." Detailed Reasoning:
* The screenshot shows a Data Page Source configuration where:
* Source = Robotic desktop automation
* Robotic automation name = FETCH_SCORE
* Timeout = 5 seconds
* Request Data Transform = Request
* Response Data Transform = RDAResponse
* The Robotic automation name (FETCH_SCORE) identifies which Robot Studio activity should execute when the Pega Platform requests the data.
* In Robot Studio, this is mapped to the ActivityName property of the automation's activity component, ensuring a direct call mapping between Pega and Robot Studio.
Option Analysis:
* A. ActivityName: Correct - this is the property used to link the automation in Pega Platform to the robotic activity in Robot Studio.
* B. FullName: Incorrect - FullName refers to the namespace or project structure reference, not the callable identifier.
* C. (Name): Incorrect - This refers to the internal object label, not the platform reference name.
* D. UniqueID: Incorrect - This is an internal system identifier used by Robot Studio for distinguishing components, not for integration with Pega Platform.
Therefore, the correct answer is A. ActivityName.
Reference:Extracted and verified from Pega Robotics System Design and Implementation Guide, Robot Activity Configuration and Pega Platform Integration (RDA/RPA) section (Pega Robotics 19.1 and later).

NEW QUESTION # 69
A developer adds a diagnostic log component to an automation and checks the log file.
Based on the image, how did the developer configure the diagnostic log component settings?

- A. Exhibit D
- B. Exhibit A
- C. Exhibit C
- D. Exhibit B

Answer: D

NEW QUESTION # 70
You create a RcboticBanking project containing a BankerInsight application, two automations (AddCustomer and RemoveCustomer). and two BankerInslght's application methods (CustomerSearch and Login).
Which option shows the Project explorer with the content defined above7

- A.
- B.
- C.
- D.

Answer: B

Explanation:
In Pega Robot Studio, the Project Explorer provides a structured view of all the components that belong to a project. These include applications, automations, and any methods (application-level procedures) associated with each application. The hierarchy in the Project Explorer reflects the relationship and scope of these components.
According to the Pega Robotics System Design and Implementation Guide (Project Structure and Scoping Rules section):
"Each application added to a Pega Robotics project appears under the 'Globals' node.
Automations are listed as separate automation objects under the application node.
Application methods such as Login or CustomerSearch are defined directly under the application to which they belong, following the hierarchy:
Globals # ApplicationName # Automations # Application Methods."
Based on this structure:
* The Globals node appears at the top.
* The BankerInsight application appears under Globals.
* Within BankerInsight, the two automations - AddCustomer and RemoveCustomer - are listed.
* Under the same application node, the application methods CustomerSearch and Login also appear, showing they belong specifically to the BankerInsight scope.
Option C correctly represents this hierarchy because it shows:
* Globals at the root.
* BankerInsight as the single application node.
* Under BankerInsight, both automations (AddCustomer, RemoveCustomer) and the application methods (CustomerSearch, Login) appear in sequence.
Other options show misplaced or missing elements:
* Option A: Incorrect - does not display both automations and methods under the same hierarchy.
* Option B: Incorrect - application methods are misplaced outside of the BankerInsight node.
* Option D: Incorrect - application methods appear incorrectly scoped under separate folders.
Therefore, Option C aligns perfectly with the standard Pega Robotics Project Explorer hierarchy for the given configuration.
References:Extracted and verified from Pega Robotics System Design and Implementation Guide,
"Project Explorer Hierarchy and Application Scoping" section (Pega Robotics 19.1 and later).

NEW QUESTION # 71
our project for a customer service department contains a Windows form with a btnUpdateAccount button.
Users click btnUpdateAccount to automate the updates of other customer account systems at the end of the call. You create the UpdateAccount automation to ensure that the Windows form is still accessible after clicking the button.
Which option represents the UpdateAccount automation with this requirement?

- A.
- B.
- C.
- D.

Answer: A

Explanation:
* Pega Robotics Studio - Automation Design Concepts (Events and UI Responsiveness)
"Automations started from a Windows Form Click event run on the UI thread. To keep the form responsive, long-running work should be started by calling other automations asynchronously. When an automation is executed synchronously, the UI thread is blocked until the call completes. Executing the child automation asynchronously allows users to continue interacting with the form."
* Pega Robotics Studio - Calling Automations (Run method)
"The Run method includes a synchronous parameter.
True - the caller waits for completion (blocks the UI).
False - the automation starts asynchronously and control returns immediately to the caller (UI remains available).
Default follows the project setting."
* Pega Robotics Studio - Windows Form Controls (Avoid self-triggering)
"Invoking PerformClick from within an automation that is already handling the button's click should be avoided. It re-triggers the button click and can lead to reentrancy or recursion and does not improve UI responsiveness."
* Pega Robotics Studio - Message Dialogs
"Displaying a MessageDialog during processing is modal and prevents interaction with the form until the dialog is closed. Use only for completion or error notifications, not while long-running work is executing." Why Option B is correct:
* Option B starts from the btnUpdateAccount.Click event (so no self-trigger via PerformClick).

* It launches the downstream automations (UpdateBankerInsight and UpdatePegasFinance) using Run with the synchronous parameter set to False (asynchronous), which keeps the Windows form responsive and accessible to the user while updates run.
* It does not introduce a modal MessageBox before or during the updates (dialogs are only used for completion/notification), so it avoids blocking the UI.
Why the other options are not correct:
* Option A: Uses PerformClick on the button, which re-triggers the click and can lead to recursion without improving responsiveness.
* Option C: Inserts a MessageDialog during the middle of processing, which is modal and blocks the form.
* Option D: Calls the update automations synchronously (or leaves them at the blocking default), which holds the UI thread until completion and makes the form inaccessible during the run.

NEW QUESTION # 72
.....

myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes