



Free Questions for UiPath-ASAPv1 by go4braindumps

Shared by Hobbs on 22-07-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

What is a critical security and compliance check performed by the Solution Architect during code reviews?

Options:

- A- Check client geographic datastore requirements.
- B- Check for Personally Identifiable Information written as clear text in logs.
- C- Check for workflows invoked with Isolated option.
- D- Check the applications used by the automation.

Answer:

B

Explanation:

Ensuring the security and compliance of automation solutions is paramount, especially when dealing with sensitive data. A critical security and compliance check performed by Solution Architects during code reviews is verifying that Personally Identifiable Information

(PII) is not written as clear text in logs. Writing PII to logs can lead to security breaches and non-compliance with data protection regulations such as GDPR. Solution Architects must ensure that the code is designed to either mask or exclude PII from logs, thereby protecting sensitive information and adhering to compliance standards.

UiPath Security Guide: Logging Best Practices

UiPath Security and Compliance: Data Protection and Privacy

Question 2

Question Type: MultipleChoice

What role must approve the Pull Requests into the Main Branch?

Options:

- A- RPA Developer Lead
- B- Client Technical Lead
- C- Solution Architect

D- CoE Lead

Answer:

C

Explanation:

In the context of UiPath and RPA development, the Solution Architect plays a critical role in ensuring that the automation projects adhere to best practices, architectural standards, and organizational policies. One of the responsibilities of the Solution Architect is to review and approve Pull Requests into the Main Branch of the project repository. This step is crucial for maintaining code quality, ensuring that changes are aligned with the overall solution design, and preventing potential issues in production environments. The approval process by the Solution Architect helps in safeguarding the integrity and reliability of automation solutions.

UiPath Best Practices: Role of a Solution Architect

UiPath Studio Guide: Version Control Systems

Question 3

Question Type: MultipleChoice

What is the benefit of High Availability?

Options:

- A-** High availability provides redundancy and stability for a multi-node Orchestrator deployment through a retrieval mechanism.
- B-** High availability provides redundancy and stability for a multi-node Orchestrator deployment through failure resistance.
- C-** High availability provides redundancy and stability for a single-node Orchestrator deployment through failure resistance.
- D-** High availability provides redundancy and stability for a single-node Orchestrator deployment through exception handling.

Answer:

B

Explanation:

High Availability (HA) in the context of UiPath Orchestrator is a setup designed to ensure that the system remains available and operational, even in the event of hardware failures, software crashes, or other unexpected issues. HA is achieved by implementing a multi-node Orchestrator deployment, where multiple instances of Orchestrator are run in parallel. This setup provides redundancy, meaning if one node fails, the others can take over, ensuring continuous operation. The key benefit of HA is its ability to resist failures and maintain stability, which is crucial for critical enterprise operations relying on RPA.

Question 4

Question Type: MultipleChoice

How can a user set up a Calendar for Non-Working Days in UiPath Orchestrator?

Options:

- A-** Create a new calendar, and manually select Non-Working Days or upload a .csv file containing Non-Working Days.
- B-** Create a new calendar, and manually select Non-Working Days.
- C-** Create a new calendar, and manually select Non-Working Days or upload a .txt file containing Non-Working Days.
- D-** Orchestrator does not have the capability to create Calendars.

Answer:

A

Explanation:

In UiPath Orchestrator, calendars are used to define working and non-working days for robots. To accommodate non-working days, such as weekends or holidays, a user can create a new calendar and manually mark these days. Additionally, for ease and efficiency, a .csv file containing all non-working days can be prepared and uploaded to Orchestrator. This feature is particularly useful for planning and scheduling unattended robots, ensuring that they operate according to the organization's working calendar.

UiPath Orchestrator Guide: Managing Calendars

UiPath Orchestrator Guide: About Calendars

Question 5

Question Type: MultipleChoice

What is the primary purpose of the "EditGrid" in UiPath Apps?

Options:

- A-** To facilitate better data management and provide users with the ability to edit and update data directly within the app.
- B-** To enable users to build interactive dashboards and advanced data visualizations using a drag-and-drop interface without coding.
- C-** To optimize communication protocols between various UiPath components.

D- To automate data extraction, transformation, and storage processes without relying on external RPA components or integrations.

Answer:

A

Explanation:

The 'EditGrid' component in UiPath Apps is designed to enhance data interaction within an app. It allows users to view, edit, and manage data in a tabular format directly within the app interface. This functionality is crucial for apps that require data manipulation, as it enables end-users to make real-time updates and changes to the data without the need for external tools or interventions. The EditGrid fosters a more dynamic and interactive user experience, making it easier to handle data-intensive tasks within UiPath Apps.

UiPath Apps Guide: EditGrid Component

UiPath Forum: How to Use EditGrid in UiPath Apps

Question 6

Question Type: MultipleChoice

Consider a process that is scheduled to run 3 times a day (9AM, 1PM, 6PM). For each run, the process needs to consume data from a different queue.

What is the best way to implement this functionality?

Options:

- A- Use the queue name as an argument for the process and set its value when creating the triggers.
- B- Create an asset which stores the queue name and update the asset value after each run.
- C- Publish the process 3 times in Orchestrator, specifying the corresponding queue in code.
- D- Add the queue name in a configuration file and store it in a shared location.

Answer:

A

Explanation:

In UiPath Orchestrator, using the queue name as an argument for the process is the most efficient way to handle varying data sources for different schedule times. When scheduling the process, you can specify different queue names as input arguments for each trigger (9AM, 1PM, 6PM). This approach allows for a single, flexible process that adapts to different data sources at different times without the need for multiple deployments or manual updates to assets or configuration files.

[UiPath Orchestrator Guide: Managing Triggers](#)

[UiPath Orchestrator Guide: About Arguments](#)

To Get Premium Files for UiPath-ASAPv1 Visit

<https://www.p2pexams.com/products/uipath-asapv1>

For More Free Questions Visit

<https://www.p2pexams.com/uipath/pdf/uipath-asapv1>

