

Pass Guaranteed Saviynt - SCAIP - High-quality Hottest Saviynt Certified Advanced IGA Professional (Level 200) Certification



The PDFVCE is one of the top-rated and renowned platforms that has been offering real and valid Saviynt Certified Advanced IGA Professional (Level 200) (SCAIP) exam practice test questions for many years. During this long time period countless Saviynt Certified Advanced IGA Professional (Level 200) (SCAIP) exam candidates have passed their dream certification and they are now certified Saviynt professionals and pursuing a rewarding career in the market.

They work together and put all their efforts to ensure the top standard of Saviynt SCAIP exam practice test questions. The SCAIP exam practice test questions are being offered in three different formats. These Saviynt SCAIP Exam Questions formats are PDF dumps files, desktop practice test software, and web-based practice test software.

>> Hottest SCAIP Certification <<

Reliable SCAIP Dumps & Valid SCAIP Exam Question

Our SCAIP practice questions attract users from all over the world because they really have their own charm. No product like our SCAIP study guide will seriously consider the needs of users in all aspects. From product content to system settings, we will give you what you want! Firstly, you definitely want to pass the exam for sure. Our SCAIP Exam Questions are high-effective with a high pass rate as 98% to 100%. So don't hesitate, just come and buy our SCAIP learning braindumps!

Saviynt Certified Advanced IGA Professional (Level 200) Sample Questions (Q37-Q42):

NEW QUESTION # 37

Which rule type is primarily used to provision birthright access, also known as zero-day provisioning, based on specified conditions?

- A. Technical Rule
- B. SAV Role
- C. Scan Rule
- D. User Update Rule

Answer: A

Explanation:

The correct answer is B. Technical Rule . Saviynt documentation clearly states that a Technical Rule is primarily used to provision birthright access , also referred to as zero-day provisioning , to users joining the organization based on specified conditions. This is one of the most important distinctions in the Rules and Policies section of the Level 200 syllabus. Technical Rules are intended for automated access assignment logic, especially where access must be granted immediately when user attributes match business conditions such as department, location, or cost center.

The other options are not correct for this use case. User Update Rules are generally used to take actions when user records change and can support lifecycle events, but the documentation identifies Technical Rules as the primary mechanism for birthright provisioning. Scan Rules are used for detection and policy-based scanning use cases, not default access assignment. SAV Role controls platform authorization inside Saviynt rather than provisioning target application access. Saviynt also documents that entitlement assignments in a Birthright Rule can be parameterized using user attributes, which reinforces that Technical Rules are the intended framework for this type of zero-day access automation.

NEW QUESTION # 38

An EIC Administrator wants to retrieve a report of users and their assigned SAV Roles. What are the ways in which it can be achieved? (Multi-Select)

- A. Run a SQL Query to retrieve the data in Data Analyzer
- B. Create analytics with the SQL Query to get the data and select the checkbox 'Send Email As Attachment' in analytics configuration
- C. Create an analytics with the SQL Query to get the data and export as CSV/Excel
- D. Using enhanced query

Answer: A,B,C

NEW QUESTION # 39

Which of the following scenarios are True to trigger Technical Rule Execution in EIC? (Multi-Select)

- A. When a user is deleted and the condition in the rule matches
- B. The existing user is updated and satisfies a user update rule with action as Re-run provisioning rules
- C. When users are imported through Import Job and the condition in the rule matches
- D. A new user is registered or created from the UI and the condition in the rule matches

Answer: B,C,D

Explanation:

In Saviynt EIC, Technical Rules are triggered based on lifecycle events related to user creation, updates, and imports, provided the defined conditions evaluate to true. The correct answers are B, C, and D.

Option B is correct because during Import Jobs, when users are brought into Saviynt from authoritative sources, Technical Rules are evaluated, and if conditions match, they are executed. This is a common mechanism for provisioning access during onboarding.

Option C is also correct since when a new user is created via the UI, Technical Rules can be triggered if the user attributes meet the rule conditions. This ensures consistent provisioning regardless of how users are created.

Option D is correct because when an existing user is updated, and a User Update Rule is configured to run provisioning rules, it can trigger associated Technical Rules again.

Option A is incorrect because deletion events typically trigger deprovisioning workflows rather than standard Technical Rule execution.

Thus, Technical Rules are triggered during import, creation, and update events-not deletion.

NEW QUESTION # 40

In REST connector, under which JSON do you specify the provisioning limit and connection timeout settings?

- A. MODIFYUSERDATAJSON
- B. StatusthresholdConfig
- C. ConfigJSON
- D. ConnectionJSON

Answer: D

Explanation:

In Saviynt EIC REST connector configuration, ConnectionJSON is the central configuration block where all connection-level properties are defined. This includes not only the base URL, authentication details, and headers, but also critical operational parameters such as connection timeout settings and provisioning limits (such as throttling or API limits).

Provisioning activities like account creation, update, and deletion rely on stable connectivity to the target system. Therefore, timeout configurations (for example, connection timeout and read timeout) are defined in ConnectionJSON to ensure that API calls do not hang indefinitely and can fail gracefully if the target system is unresponsive. Similarly, provisioning limits-such as maximum records processed or API throttling controls-are configured at this level to manage performance and avoid overloading external systems.

Option A (ConfigJSON) is not a standard JSON used in REST connector configurations. Option B (MODIFYUSERDATAJSON) is used specifically for user update operations. Option D (StatusthresholdConfig) relates to job/status handling rather than connection parameters.

Thus, ConnectionJSON is the correct place for configuring provisioning limits and timeout settings in Saviynt REST connectors.

NEW QUESTION # 41

Choose the correct SQL query from the below options to populate an attribute with the logged-in user's Display Name

- A. select DISPLAYNAME as ID from Users u where u.userkey=\${loggedInUser}
- B. select DISPLAYNAME as ID from Users u where u.userkey=\${requestor}
- C. All the above
- D. select DISPLAYNAME as ID from Users u where u.userkey=\${user.id}

Answer: A

Explanation:

In Saviynt EIC, when configuring dynamic attributes or custom forms, system variables are used to fetch context-specific data such as the currently logged-in user. To retrieve the Display Name of the logged-in user, the correct variable is \${loggedInUser}, which directly represents the userkey of the active session user.

Option A correctly uses this variable in the SQL query to fetch the DISPLAYNAME from the Users table.

This ensures that the attribute dynamically reflects the logged-in user's display name at runtime.

Option B is incorrect because \${user.id} typically refers to the target user in a request context, not necessarily the logged-in user.

Option C (\${requestor}) may represent the requester in certain workflows but is not consistently equivalent to the logged-in user in all scenarios, especially in delegated or admin-driven requests.

Therefore, Option A is the most accurate and reliable approach for retrieving the logged-in user's display name in Saviynt configurations, ensuring proper context-aware data population.

NEW QUESTION # 42

.....

- [illegible]