TESTPASSPORT

ÜBUNGSFRAGEN

HOCHWERTIGE LERNRESSOURCEN

Exam : **250-613**

Title : AutoSys Technical Specialist

Version : DEMO

1.In an AutoSys High Availability (HA) Dual Event Server deployment, which of the following statements correctly describe the behavior and configuration of the core components? (Select all that apply)
A. Both Event Server 1 and Event Server 2 process read and write operations simultaneously in an active-active cluster to balance the load.
B. The Scheduler is configured with connection details for both Event Servers but actively writes only to the primary database until a failover occurs.
C. If the primary Event Server fails, the Scheduler automatically rolls over to the secondary Event Server without requiring a manual restart.
D. The Application Server (App Server) caches all database transactions and manually syncs them to the secondary Event Server when the primary goes down.
Answer: B, C
Explanation:
● Option B is correct. In a Dual Event Server setup, the Scheduler is aware of both databases. However, it operates in an active-passive data writing model regarding the Event Servers, primarily interacting with the active database.
● Option C is correct. The core feature of AutoSys HA is the Scheduler's ability to detect a database failure and automatically "roll over" to the secondary database to continue processing workload events seamlessly.
● Option A is incorrect. AutoSys Dual Event Servers operate in an active-passive redundancy mode for the application level, not as an active-active load-balancing database cluster.
● Option D is incorrect. The App Server does not cache and sync database transactions; it relies on the database synchronization mechanisms (usually configured at the RDBMS storage level) to ensure data consistency between the primary and secondary databases.

2.You are writing a JIL script to create a command job that executes a Python script. Based on standard AutoSys job types, provide the exact value that should be assigned to the `job_type` attribute.

```
insert_job: RUN_DATA_MODEL
job_type: ___
machine: LINUX_PROD_01
command: /usr/bin/python3 /scripts/model.py
owner: data_user
```

Answer: cmd (or c)
Explanation:
● For jobs that execute a standard operating system command, shell script, or executable, the required `job_type` in JIL is `cmd` (or its abbreviation `c`).

3.A nightly database cleanup job `DB_CLEANUP_NIGHTLY` failed during its run due to a temporary network glitch and is now in a `FAILURE` status. The network issue is resolved, and as the AutoSys Administrator, you need to execute the job again immediately, bypassing its normal time schedule. Which command should you use?
A. `sendevent -E RESTART -J DB_CLEANUP_NIGHTLY`
B. `sendevent -E FORCE_STARTJOB -J DB_CLEANUP_NIGHTLY`

C. `sendevent -E STARTJOB -J DB_CLEANUP_NIGHTLY`
D. `sendevent -E CHANGE_STATUS -s SUCCESS -J DB_CLEANUP_NIGHTLY`
Answer: B
Explanation:
● Option B is correct. `FORCE_STARTJOB` is the correct event to forcefully trigger a job to run immediately, regardless of its current status (`FAILURE`, `SUCCESS`, etc.) and bypassing any starting conditions or time schedules it normally waits for.
● Option A is incorrect. `RESTART` is not a valid standard event in AutoSys for this context.
● Option C is incorrect. `STARTJOB` will only start the job if its defined starting conditions (like time or dependencies) are currently met. Since the normal schedule window might have passed, it may not run.
● Option D is incorrect. `CHANGE_STATUS` to `SUCCESS` merely alters the logical state in the database; it does not actually execute the cleanup script on the remote agent.

4.You are designing a dependency workflow for `JOB_C`. You want `JOB_C` to run ONLY IF `JOB_A` completes successfully AND `JOB_B` fails.
Which of the following JIL configurations correctly implement this starting condition? (Select all that apply)
A. `condition: success(JOB_A) & failure(JOB_B)`
B. `condition: s(JOB_A) and f(JOB_B)`
C. `condition: s(JOB_A) & f(JOB_B)`
D. `condition: (success(JOB_A) | failure(JOB_B))`
Answer: A, C
Explanation:
● Option A is correct. This uses the full keywords `success` and `failure` along with the standard logical AND operator `&`.
● Option C is correct. This uses the standard accepted shorthand `s` for success and `f` for failure, combined with `&`. AutoSys JIL parsers interpret A and C identically.
● Option B is incorrect. While some programming languages use `and`, AutoSys JIL strictly requires the ampersand `&` for the logical AND operator in conditions.
● Option D is incorrect. The pipe `|` represents a logical OR. This configuration would start `JOB_C` if *either* condition were true, which violates the strict requirement.

5.When integrating AutoSys Workload Automation with Embedded Entitlements Manager (EEM) for security, how are permissions primarily structured to grant a specific group of users the ability to execute, but not modify, jobs matching the naming convention `FIN_*`?
A. By modifying the local UNIX `chmod` permissions on the JIL text files before they are imported.
B. By configuring an EEM Policy tied to the `as-job` resource class, assigning an execution action to the target user group, and specifying the resource filter as `FIN_*`.
C. By setting the `owner` attribute in the JIL definition of every `FIN_*` job to the name of the user group.
D. By editing the Agent configuration file (`agentparm.txt`) to restrict command execution exclusively for financial users.
Answer: B
Explanation:
● Option B is correct. EEM manages AutoSys access control via centralized Policies based on Resource

Classes. To control access to jobs, administrators create policies against the `as-job` (AutoSys Job) resource class. You can assign specific actions (like execute, read) to user identities or groups, and scope those actions using wildcards (`FIN_*`) in the resource filter.

● Option A is incorrect. Modifying text file permissions has no bearing on how the AutoSys database and Web UI enforce runtime security.

● Option C is incorrect. The `owner` attribute typically specifies the OS-level account under which the script executes on the remote machine, not the RBAC group managing it in the UI/CLI.

● Option D is incorrect. Agent configuration handles local machine security (e.g., executing commands), but the granular UI/job control policies must be managed centrally via EEM.