PassQueen
a.k.a dumpsqueen.com

SAS Data Integration Development

SAS Institute A00-260

Version Demo

Total Demo Questions: 9

Total Premium Questions: 98

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Working with SAS Data Integration Studio	12
Topic 2, Working with Data	13
Topic 3, Working with Jobs	7
Topic 4, Working with Transformations	38
Topic 5, Loading Data	8
Topic 6, Managing and Deploying Jobs	20
Total	98

QUESTION NO: 1

Which statements are true about registering physical tables in SAS Data Integration Studio? (Choose two.)

A. It creates metadata for the physical table.

B. It enables the table to be reused in metadata-driven jobs.

C. It copies all physical table rows into the SAS Metadata Repository.

D. It converts a database table into a SAS data set.

ANSWER: A B

Explanation:

Registering a physical table creates metadata that describes the table in the SAS Metadata Repository. The metadata can include the table's columns, formats, keys, indexes, and library association. The registered table can then be found and reused in Data Integration Studio jobs and other SAS applications that use the metadata.

Registration does not copy the physical table's data into the metadata repository, and it does not require that the physical table be converted into a SAS data set. The underlying table remains in its original physical location and is accessed through its associated library definition.

See the SAS metadata folder documentation for information about metadata organization.

QUESTION NO: 2

You want to register an external file with the following structure: first line contains Customer First and Last Name second line is address1 third line is address 2 fourth line contains the phone number What action should you take to read this file?

A. Use the New User Written External File wizard.

B. Use the New Fixed Width External File wizard.

C. Use the New Delimited External File wizard.

D. Use the New COBOL Copybook wizard.

ANSWER: A

Explanation:

Use the New User Written External File wizard. The described source is a multi-line logical record: one customer’s information is spread across four successive physical lines, with each line representing a different attribute or group of attributes. Reading that structure requires custom input logic that can explicitly control how SAS reads each line and assigns its values to the appropriate columns in one output observation. The user-written external-file registration approach is intended for file layouts that cannot be adequately described by the standard delimited or fixed-width metadata definitions. It enables the developer to supply SAS code, typically using INPUT processing and appropriate line-control features, so that the source is read according to its actual record structure. This lets the resulting metadata accurately represent a single customer row containing name, both address lines, and phone number, despite the values originating on separate lines. SAS documentation describes external-file access and custom file-reading techniques in the SAS Data Integration Studio user documentation and SAS INPUT statement reference. See the SAS Data Integration Studio User’s Guide and the SAS INPUT statement documentation.

QUESTION NO: 3

Select the valid reasons for using SAS Data Integration Studio's Table Loader transformation. (Choose two.)

A. Create and replace permanent tables.

B. Maintain access privileges to permanent tables.

C. Apply and maintain constraints on permanent tables.

D. Delete and remove permanent tables.

ANSWER: A C

Explanation:

The Table Loader transformation is appropriate when a job must create a target table or replace an existing permanent target table as part of loading data. Its generated load process supports table-creation and replacement behavior, allowing the target's physical definition to be deployed consistently from the table metadata. This makes it useful for repeatable ETL jobs in which the target structure must be created or refreshed before data is loaded.

It is also appropriate for applying and maintaining constraints on permanent tables. Constraints defined for a target table, such as integrity constraints, are part of the table's metadata and can be included when the target table is created or maintained through the load process. Maintaining these definitions with the target helps ensure that the table enforces the intended data-integrity rules after deployment and during subsequent job runs.

SAS documents the Table Loader as a transformation for loading a physical target table and configuring the target-table loading behavior. The Data Integration Studio documentation also describes how table metadata, including constraints, is managed and deployed. See the SAS Data Integration Studio User's Guide and the SAS Data Integration Studio User's Guide PDF.

QUESTION NO: 4

In an SCD Type 2 dimension table, what is the purpose of a current-row indicator column?

A. It identifies the currently effective version of a dimension member.

B. It identifies the primary key column of the source table.

C. It identifies rows that failed data validation.

D. It identifies the next surrogate key value to assign.

ANSWER: A

Explanation:

A current-row indicator identifies the version of a dimension row that is currently effective for a business key. When a tracked attribute changes, SCD Type 2 processing expires the prior row and inserts a new version. The new version is marked as current, while the earlier version remains available for historical reporting. See the SAS Data Integration Studio User's Guide.

QUESTION NO: 5

Which statements are true regarding a left outer join in the SQL Join transformation? (Choose two.)

A. All rows from the left input table are retained.

B. Unmatched right-table columns contain missing values for retained left rows.

C. All rows from the right input table are retained.

D. Only rows with matches in both input tables are returned.

ANSWER: A B

Explanation:

A left outer join retains every row from the left input table. When a left-table row has a matching row in the right table, columns from both tables are returned. When no right-table row matches, the right-table columns in the result contain missing values. Unmatched rows that exist only in the right input are not included. See the SAS SQL joined-table documentation.

QUESTION NO: 6

After creating a stored process from a SAS Data Integration Studio job, how can you test the stored process to make sure it runs correctly?

A. Execute the stored process in SAS Data Integration Studio.

B. Execute the stored process in SAS Management Console.

C. Execute the stored process using the SAS Add-In for Microsoft Office or SAS Enterprise Guide.

D. Execute the stored process using SAS Display Manager's Stored Process Test window.

ANSWER: C

Explanation:

Execute the stored process using the SAS Add-In for Microsoft Office or SAS Enterprise Guide. A stored process created from a SAS Data Integration Studio job is registered in SAS metadata and is intended to be invoked by a supported client application. SAS Enterprise Guide and the SAS Add-In for Microsoft Office provide user-facing interfaces for locating a stored process, supplying any prompted parameter values, submitting it to the configured SAS Stored Process Server or workspace environment, and viewing the resulting output. This makes them appropriate tools for functional testing: the developer can verify that the process resolves its metadata dependencies, accepts parameter values as designed, connects to its assigned execution server, and produces the expected results or output objects. Testing through one of these clients also approximates the way business users commonly consume stored processes after deployment. SAS documentation describes stored processes as SAS programs that are stored and managed centrally and can be executed from client applications, including SAS Enterprise Guide and the Microsoft Office add-in. See the SAS Stored Processes documentation and the SAS Enterprise Guide documentation.

QUESTION NO: 7

Assume that SAS Data Integration Studio Table Loader is being used in a job. The load style selected is Append to Existing. Which statement is true regardingAssume that SAS Data Integration Studio? Table Loader is being used in a job. The load style selected is

Append to Existing. Which statement is true regarding this load style?

A. It generates PROC APPEND code.

B. It generates PROC SQL code.

C. It generates either PROC SQL code or PROC APPEND code depending on further option choices.

D. It generates DATA STEP APPEND code.

ANSWER: C

Explanation:

“It generates either PROC SQL code or PROC APPEND code depending on further option choices.” is correct. The Table Loader transformation’s Append to Existing load style adds rows from the source to a target table that is already present, without replacing the target’s existing contents. SAS Data Integration Studio selects generated code appropriate to the target and the transformation configuration. For a SAS table, generated code can use the efficient PROC APPEND procedure, which adds observations to the base data set. In configurations where SQL processing is selected or is required by the

target-table access method, the generated job can instead use PROC SQL to insert the incoming rows into the existing table. Thus, Append to Existing describes the intended loading behavior, while the generated implementation can vary based on the selected load options and target environment. This flexibility lets a single Table Loader load style support both SAS-native append processing and SQL-based insert processing while preserving existing target rows. SAS documents PROC APPEND as the procedure used to append data sets and documents SQL INSERT processing as the SQL mechanism for adding rows to a table. See the SAS PROC APPEND documentation and the SAS PROC SQL documentation.

QUESTION NO: 8

In SAS Data Integration Studio, which component allows the definition of job flows with dependencies between different jobs?

A. the deployment wizard of SAS Data Integration Studio

B. the Scheduling Manager plug-in in SAS Management Console

C. the Scheduling Server

D. the SAS DATA Step Batch Server

ANSWER: B

Explanation:

The Scheduling Manager plug-in in SAS Management Console is the component used to define and manage job flows that contain dependencies among multiple deployed jobs. A job flow represents the order and conditions under which jobs run, allowing an organization to coordinate related extraction, transformation, loading, and reporting processes. In the Scheduling Manager, an administrator can add deployed jobs to a flow and specify dependency relationships so that downstream work begins only after prerequisite work has completed according to the defined flow logic. This makes the plug-in the central metadata-management interface for creating a coordinated schedule rather than merely running an individual job. The flow definition is stored in SAS metadata and can then be used with the configured scheduling environment to execute the associated jobs in the required sequence. SAS documentation describes Scheduling Manager as the SAS Management Console plug-in for scheduling deployed jobs and creating job flows. See the SAS Data Integration Studio User's Guide and the SAS Management Console documentation for scheduling and job-flow administration details.

QUESTION NO: 9

Which statements are true when the SCD Type 2 Loader transformation detects a change in a tracked dimension attribute? (Choose two.)

A. The current dimension row can be expired.

B. A new dimension-row version can be inserted.

C. The existing row is always overwritten without retaining history.

D. The business key is replaced with a new value for every change.

ANSWER: A B

Explanation:

The SCD Type 2 Loader uses the business key to identify the existing dimension member. When designated non-key attributes have changed, it preserves history by expiring the current dimension row and adding a new row for the changed version. The new row can receive a new surrogate key and be marked as the current version. See the SAS Data Integration Studio User's Guide.