DP-900: foundation worksheet

Eight selected original learning tasks reused from this foundation path. Write answers before opening the separate key. Follow each item selection instruction. This is not an unseen mock, an official examination or a readiness measure.

Independent review status is recorded in the separate bound review evidence. Qualified educator review is pending; learner trial has not run. Difficulty is editorial and uncalibrated.

Questions - separate answer key follows

01. dp900-q-001

A fictional research team is sorting five representations. Select TWO that the descriptions identify as semi-structured, rather than fixed-schema tabular data or unstructured media.

A. JSON device records whose optional property names vary from device to device

B. A CSV export with the same declared columns for every row

C. Tagged event records whose permitted payload fields vary by event type

D. Audio recordings stored as binary media files

E. A SQL table whose rows use the same declared typed columns

Your answer: _____ Reasoning notes:

02. dp900-q-018

A complete fictional Clubs table has unique ClubId values C1 and C2. Sessions(SessionId, ClubId) has (S1,C1), (S2,C1), (S3,C2), (S4,C3). This exercise intentionally includes the unmatched C3 value and assumes no foreign-key constraint rejects it. How many rows result from an INNER JOIN of Clubs to Sessions ON Clubs.ClubId = Sessions.ClubId, with no DISTINCT or other filtering?

A. 8 rows

B. 2 rows

C. 4 rows

D. 3 rows

Your answer: _____ Reasoning notes:

Questions - separate answer key follows

03. dp900-q-025

Two fictional applications must retain their current storage interfaces without an application rewrite. A requires an SMB shared-file endpoint. B already issues Azure Blob REST API requests to store and retrieve media objects. Select TWO proposals that match the respective stated interfaces.

A. Use Azure Files over SMB for application A

B. Use Azure Table Storage as application A’s existing SMB share

C. Use Azure Blob Storage for application B’s existing Blob REST requests

D. Use a Blob NFS interface as application A’s unchanged SMB endpoint

E. Use an Azure Files endpoint for application B’s unchanged Blob REST requests

Your answer: _____ Reasoning notes:

---

04. dp900-q-034

A fictional analytics project chooses its primary source collection before the future analysis schemas are known. That collection must expose mixed source files in their original formats; each later analysis chooses how to parse them. Which proposed organization directly fits that access and processing contract? All proposals retain source information somewhere.

A. Store the original files in a data lake and parse the required fields in later analysis steps

B. Parse sources into a fixed warehouse-table schema before loading the primary collection, and keep file copies separately

C. Prepare dimensional aggregate models before loading the primary collection, and keep source copies separately

D. Extract all source content into normalized operational tables as the primary collection, and retain export copies

Your answer: _____ Reasoning notes:

Questions - separate answer key follows

05. dp900-q-050

In a fictional Cosmos DB for NoSQL single-region-write workflow, two clients read item version E1. Client A conditionally replaces the item and it becomes E2. Client B must avoid silently overwriting A's change with its old calculation. Which next request-and-response interpretation fits optimistic concurrency?

A. Keep the local E1 value but compare it only with the E1 value in B’s original cache

B. Send B’s replacement with If-Match E1; treat a precondition failure as a reason to reread and reconcile

C. Read E2 and attach it to B’s unchanged old calculation without reviewing the changed fields

D. Send B’s replacement without a version condition and use the write success as evidence that no concurrent edit occurred

Your answer: _____ Reasoning notes:

06. dp900-q-055

A fictional reviewer sees a columnar Parquet layout and predicts that every query will be faster than a row-oriented alternative. The actual workload includes narrow analytical scans and frequent small point updates. Which evaluation best addresses that prediction?

A. Reject all analytical use of Parquet because one point-update path has additional work

B. Choose the layout from total row count alone and omit which columns and operations are accessed

C. Treat columnar layout as relevant to selective column reads, then measure the actual scan and update workloads rather than infer a universal winner

D. Approve the prediction after comparing one narrow scan because the remaining operations use the same file extension

Your answer: _____ Reasoning notes:

Questions - separate answer key follows

07. dp900-q-005

A complete fictional CSV export is described by four lines: header ClientId, Units; then C1,3; then C2,7; then C3,0. Each data line is one client record. How many client records does this export contain?

A. 4 records

B. 10 records

C. 3 records

D. 2 records

Your answer: _____ Reasoning notes:

---

08. dp900-q-022

Select TWO offerings described as managed SQL Server PaaS database platforms, where underlying OS/database maintenance is delegated as part of that platform rather than a guest VM being administered by the customer.

A. Azure SQL Database

B. SQL Server on a customer-administered Azure VM

C. SQL Server on the organization’s own physical server

D. Azure SQL Managed Instance

E. PostgreSQL on a customer-administered Azure VM

Your answer: _____ Reasoning notes:

Answer key and original explanations

01. dp900-q-001

A. JSON device records whose optional property names vary from device to device

C. Tagged event records whose permitted payload fields vary by event type

1. Classify the representation described, rather than deciding from the product name alone. The varying optional properties and event-specific payload fields both provide organization without one fixed set of fields for every record.
2. The declared CSV columns and SQL columns define fixed tabular schemas in these descriptions. Audio recordings are the unstructured media example. JSON can be used with a fixed schema too; the stated field variation controls this case.
3. Check a changed condition: if the event contract instead required exactly the same declared fields on every record, variable payloads would no longer be the stated reason to call it semi-structured. The answer is tied to the descriptions, not to a permanent ranking of file extensions. Audio metadata can be tabular without making the audio itself a fixed-schema record.

Why not B: The CSV description explicitly fixes the columns across rows; human-readable text does not by itself make that representation semi-structured.

Why not D: The audio description supplies media content rather than field-bearing records with a flexible organization.

Why not E: The SQL table description uses the same declared columns for each row, which is the fixed-schema contrast.

---

02. dp900-q-018

D. 3 rows

1. Enumerate matching pairs under the equality condition. C1 matches S1 and S2, giving two output rows; C2 matches S3, giving one. S4 has no matching club and contributes none.
2. The result has 2 + 1 = 3 rows. Counting clubs gives 2, retaining every session gives 4, and dropping the join condition gives a 2 × 4 Cartesian product of 8.
3. Enumerate pairs rather than source-table sizes: C1 matches S1 and S2, C2 matches S3, and the intentionally unmatched S4/C3 has no club partner. Three matching pairs remain. The absence of a rejecting foreign-key constraint permits S4 in the source but does not make it join successfully. A validity rule on stored data and a condition selecting result pairs are different mechanisms.

Why not A: Ignoring the ON condition creates every club/session pair: 2 × 4 = 8, not the specified equality inner join.

Why not B: Counting the two parent clubs discards the second matching session for C1. The query has no DISTINCT.

Why not C: Counting all four sessions wrongly keeps S4, whose C3 value has no parent match in this intentionally unconstrained exercise.

Answer key and original explanations

03. dp900-q-025

A. Use Azure Files over SMB for application A

C. Use Azure Blob Storage for application B’s existing Blob REST requests

1. Keep the two interfaces separate: A needs SMB file sharing; B needs the Blob REST interface for object operations. Azure Files addresses the former, while Blob Storage addresses the latter.
2. A file-like capability is not automatically the same wire protocol. Blob Storage supports NFS in applicable configurations, but that does not turn NFS into the required SMB endpoint. Likewise, an Azure Files REST endpoint is not the unchanged Blob REST interface.
3. Test each application's actual endpoint rather than choosing one service for the broad word files. SMB is the first unchanged interface; Blob REST is the second. A supported alternative protocol does not automatically accept the requests already sent by either client. If an application rewrite were allowed, the candidates might change, but the present comparison intentionally fixes the client contracts rather than comparing all storage uses.

Why not B: Table Storage exposes a key/attribute NoSQL model, not the existing SMB file-share interface required by A.

Why not D: NFS and SMB are different protocols. A explicitly requires SMB without rewriting the application.

Why not E: Files can expose REST access, but B specifically retains Blob REST requests; generic REST support does not make the service APIs interchangeable.

---

04. dp900-q-034

A. Store the original files in a data lake and parse the required fields in later analysis steps

1. The contract defers analysis-specific parsing and keeps the primary source collection file-oriented. A lake of original files directly provides that organization.
2. The other plans can preserve information, but make a fixed relational, dimensional or operational structure the primary access path before the stated later analyses. This does not claim warehouses can never retain raw data; it compares these supplied processing contracts.
3. The primary access contract is original files with analysis-specific parsing deferred. A fixed dimensional model could be useful later, but making it the initial required structure imposes a decision the stem postpones. The alternatives are not rejected for destroying all information; they are rejected for changing when and how readers must interpret it. A sound explanation should compare that contract rather than claim warehouses can never hold source data.

Why not B: This plan imposes the primary warehouse schema before loading, whereas the given collection defers analysis-specific parsing.

Why not C: An aggregate model chooses a reporting structure before the unspecified future analyses; it is not the required original-file primary collection.

Why not D: The primary access path becomes operational tables rather than the required original-format file collection.

Answer key and original explanations

05. dp900-q-050

B. Send B’s replacement with If-Match E1; treat a precondition failure as a reason to reread and reconcile

1. The comparison must involve the current server item. B's If-Match E1 condition no longer matches E2, so the conditional write is rejected with a precondition failure rather than silently replacing the newer version. The client should fetch the current item and decide how to reconcile the business change.
2. Merely obtaining the new tag and applying an old calculation defeats the intended lost-update protection. Likewise, an unconditional successful write only says that a write was accepted; it does not show that B calculated from the latest state. The stem limits this case to the described single-region workflow rather than generalizing every multi-region conflict policy.

Why not C: Refreshing a token without recomputing or reconciling the old business update can still discard A’s changed fields.

Why not D: Omitting the condition removes the stated version check; success does not certify that no intervening write happened.

Why not A: Two cached copies of the same old value do not test the server’s current version.

---

06. dp900-q-055

C. Treat columnar layout as relevant to selective column reads, then measure the actual scan and update workloads rather than infer a universal winner

1. A columnar representation can be relevant when a reader needs a few columns, but the format fact is not a benchmark for all operations. The two workloads touch data differently. Specify representative queries, required updates, correctness and resource conditions before comparing actual behavior.
2. One favorable scan cannot establish every-query superiority, just as one expensive update cannot establish that the format is unsuitable for every analytic scan. The comparison needs the access pattern and measured outcomes. This question makes no Azure cost or latency claim and does not convert a layout characteristic into a guarantee.

Why not D: It extrapolates one operation and sample to a different update workload without measuring that work.

Why not A: It turns a limitation for one operation into an equally unsupported rejection of a different workload.

Why not B: Row count alone omits column selection, filtering and update behavior, which are central to the prediction.

Answer key and original explanations

07. dp900-q-005

C. 3 records

1. Separate the header from the data lines. C1, C2 and C3 are the three records; the zero Units value does not remove C3.
2. Counting the header gives4. Adding Units gives3+7+0=10 units rather than a record count. Counting the two column names gives2 columns.
3. Mark the header as schema information, then enumerate C1, C2 and C3 as the three records. C3's zero Units is still a supplied client record; omitting it changes the population from clients to clients with positive units. Counting four would instead mix one header line with three data records. This is an explicit simple export model, not a claim that every physical CSV line is a complete record.

Why not A: This counts the header as though it were another client.

Why not B: This sums Units,3+7+0=10, which is not the requested count.

Why not D: This counts the two fields, not the three data lines.

---

08. dp900-q-022

A. Azure SQL Database

D. Azure SQL Managed Instance

1. SQL Database and Managed Instance provide managed SQL Server database platforms. Their common maintenance boundary does not mean their supported feature scopes are identical.
2. Running database software on a self-administered VM or physical host is a different operating model. Adding the word Azure to that host does not make it one of these two managed SQL Server PaaS offerings.
3. Keep the responsibility boundary explicit: delegating underlying maintenance does not delegate every data definition, access decision, query design or business recovery requirement. The two named PaaS offerings satisfy the question's managed-platform description, while the customer-administered guest VM does not. A service supporting the SQL Server engine can belong to a different operating model; matching the engine name alone does not settle who maintains its host.

Why not B: The stated VM is customer-administered; it is not the managed database-platform boundary in the question.

Why not C: A customer physical server does not supply the requested Azure PaaS platform.

Why not E: The option explicitly specifies a customer-administered VM and a different engine.

Sources and learning scope

Selected foundations; all 30 retained official leaves remain partial. Additional evidence is not a full-outline or calibrated simulation claim.

Questions, answer options and explanations are original teaching content. The references support selected concepts; this worksheet does not establish complete official coverage or professional readiness.

Explore analytical data processing - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-core-data-concepts/6-analytical-processing

What Is Azure SQL? - Azure SQL | Microsoft Learn
https://learn.microsoft.com/en-us/azure/azure-sql/azure-sql-iaas-vs-paas-what-is-overview?view=azuresql

Identify data formats - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-core-data-concepts/2-data-formats

Explore file storage - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-core-data-concepts/3-file-storage

Understand normalization - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-relational-data-offerings/3-normalization

Understand relational data - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-relational-data-offerings/2-understand-relational-data

Explore SQL - Training | Microsoft Learn
https://learn.microsoft.com/en-us/training/modules/explore-relational-data-offerings/4-query-with-sql

Introduction to Azure Storage - Cloud storage on Azure | Microsoft Learn
https://learn.microsoft.com/en-us/azure/storage/common/storage-introduction

Microsoft Learn: rd-dp-cosmos
https://learn.microsoft.com/en-us/azure/cosmos-db/database-transactions-optimistic-concurrency