Question 1: Skipped

Synapse Studio comes with an integrated notebook experience. The notebooks in Synapse studio, are a web interface that enables you to create, edit, or transform data in the files. It is based on a live code experience, including visualizations and narrative text.

True or False: You can access data in the primary storage account directly. There's no need to provide the secret keys.

- True
  (Correct)
- False

Explanation
Synapse Studio comes with an integrated notebook experience. The notebooks in Synapse studio, are a web interface that enables you to create, edit, or transform data in the files. It is based on a live code experience, including visualizations and narrative text.

If you'd like to experiment with your data and gain some insights about the data, notebooks are a good way to start and validate some of the ideas you might have.

The look and feel of the integrated notebook experience is similar to, for example, the jupyter notebooks in Azure Machine Learning Service or other IDEs you might use and interact with on your data.

If you navigate to the Synapse studio environment, you can find the notebooks in the Development Hub of the studio experience. To access the studio environment, you can navigate to the Azure Synapse Analytics Workspace and launch the studio. You'll also find that there are some notebook examples available through the Knowledge Centre.

The notebooks allow you to write multiple languages in one notebook by using the magic commands expressed by %%

The visual aspects of the notebooks are

• Support for Language Syntax highlight

• Syntax error

• Syntax code completion

• Export results

Within the notebook environment of the Azure Synapse Analytics Studio, you have the possibility tc create temporary tables across the multiple languages you might use.

In order to ingest data through notebooks you can use a linked service from the workspace, to, for example, an Azure Data Lake storage where then the keys and access are automatically passed through to the storage account where you have stored the file that you want to ingest or read out into a spark DataFrame.

You can access data in the primary storage account directly. There's no need to provide the secret keys. In the Data tab on the left hand-side in the Synapse Workspace, right-click on a file and select New notebook to see a new notebook with data extractor autogenerated.

With an Azure Synapse Studio notebook, you can:

• Get easily started.

• Keep data secure with built-in enterprise security features.

• Analyze data across raw formats (CSV, txt, JSON, etc.), processed file formats (parquet, Delta Lake, ORC, etc.), and SQL tabular data files against Spark and SQL.

• Be productive with enhanced authoring capabilities and built-in data visualization.

https://docs.microsoft.com/en-us/azure/synapse-analytics/spark/apache-spark-development-using-notebooks?tabs=classical

Question 2: Skipped

How can parameters be passed into an Azure Databricks notebook from Azure Data Factory?

- Use the new API endpoint option on a notebook in Databricks and provide the parameter name.
- Deploy the notebook as a web service in Databricks, defining parameter names and types.
- Use notebook widgets to define parameters that can be passed into the notebook. **(Correct)**

Render the notebook to an API endpoint in Databricks, defining parameter names and types.

Explanation
You can configure parameters by using widgets on the Databricks notebook. You then pass in parameters with those names via a Databricks notebook activity in Data Factory.

https://docs.databricks.com/notebooks/widgets.html

Question 3: Skipped

Azure Data Factory provides a variety of methods for ingesting data, and also provides a range of methods to perform transformations.

These methods are:

• Mapping Data Flows

• Compute Resources

• SSIS Packages

Mapping Data Flows provides a number of different transformations types that enable you to modify data. They are broken down into the following categories:

• Schema modifier transformations

• Row modifier transformations

• Multiple inputs/outputs transformations

Which of the following are valid transformations available in the Mapping Data Flow? (Select all that apply)

- ☐ Aggregate **(Correct)**
- ☐ Round
- ☐ Filter **(Correct)**
- ☐ Exists

(Correct)

- Trim
- Conditional split
(Correct)
- Union
(Correct)
- Derived column
(Correct)
- Merge
- Flatten
(Correct)
- Join
(Correct)
- Between
- Lookup
(Correct)
- Alter row
(Correct)
- Avg

Explanation

Just as Azure Data Factory provides a variety of methods for ingesting data, it also provides a range of methods to perform transformations. You can pick a method that matches the skillsets of your team or takes advantage of existing technologies that you already have in your data estate. There is also the opportunity to perform transformations without writing code at all using the Mapping Data Flow.

Transforming data using Mapping Data Flow

Mapping Data Flows provide an environment for building a wide range of data transformations visually without the need to use code. The resulting data flows that are created are subsequently executed on scaled-out Apache Spark clusters that are automatically provisioned when you execute the Mapping Data Flow. Mapping Data Flows also provides the capability to monitor the execution of the transformations so that you can view how the transformations are progressing, or to understand any errors that may occur

Mapping Data Flows provides a number of different transformations types that enable you to modify data. They are broken down into the following categories:

• Schema modifier transformations

• Row modifier transformations

• Multiple inputs/outputs transformations

Below is a list of transformations that is available in the Mapping Data Flows:

Name & Category: Aggregate - Schema modifier

Description: Define different types of aggregations such as SUM, MIN, MAX, and COUNT grouped by existing or computed columns.

Name & Category: Alter row - Row modifier

Description: Set insert, delete, update, and upsert policies on rows. You can add one-to-many conditions as expressions. These conditions should be specified in order of priority, as each row will be marked with the policy corresponding to the first-matching expression. Each of those conditions can result in a row (or rows) being inserted, updated, deleted, or upserted. Alter Row can produce both DDL & DML actions against your database.

Name & Category: Conditional split - Multiple inputs/outputs

Description: Route rows of data to different streams based on matching conditions.

Name & Category: Derived column - Schema modifier

Description: Generate new columns or modify existing fields using the data flow expression language.

Name & Category: Exists - Multiple inputs/outputs

Description: Check whether your data exists in another source or stream.

Name & Category: Filter - Row modifier

Description: Filter a row based upon a condition.

Name & Category: Flatten - Schema modifier

Description: Take array values inside hierarchical structures such as JSON and unroll them into individual rows.

Name & Category: Join - Multiple inputs/outputs

Description: Combine data from two sources or streams.

Name & Category: Lookup - Multiple inputs/outputs

Description: Enables you to reference data from another source.

Name & Category: New branch - Multiple inputs/outputs

Description: Apply multiple sets of operations and transformations against the same data stream.

Name & Category: Pivot - Schema modifier

Description: An aggregation where one or more grouping columns has distinct row values transformed into individual columns.

Name & Category: Select - Schema modifier

Description: Alias columns and stream names, and drop or reorder columns.

Name & Category: Sink – N/A

Description: A final destination for your data.

Name & Category: Sort - Row modifier

Description: Sort incoming rows on the current data stream.

Name & Category: Source – N/A

Description: A data source for the data flow.

Name & Category: Surrogate key - Schema modifier

Description: Add an incrementing non-business arbitrary key value.

Name & Category: Union - Multiple inputs/outputs

Description: Combine multiple data streams vertically.

Name & Category: Unpivot - Schema modifier

Description: Pivot columns into row values.

Name & Category: Window - Schema modifier

Description: Define window-based aggregations of columns in your data streams.

https://docs.microsoft.com/en-us/azure/data-factory/transform-data

Question 4: Skipped

Scenario: The company you work at stores several website asset types in Azure Storage. These types include images and videos. Which of the following is the best way to secure browser apps to lock `GET` requests?

- Lock `GET` requests down to specific domains using `CORS`.
  (Correct)
- Use Private Endpoints between the VMs and the company websites.
- Use Private Link on the company's websites.
- Lock `GET` requests down to specific domains using Vault.

Explanation
CORS support

Many companies store several website asset types in Azure Storage. These types include images and videos. To secure browser apps, it is recommended to lock `GET` requests down to specific domains.

Azure Storage supports cross-domain access through cross-origin resource sharing (CORS). CORS uses `HTTP` headers so that a web application at one domain can access resources from a server at a different domain. By using CORS, web apps ensure that they load only authorized content from authorized sources.

CORS support is an optional flag you can enable on Storage accounts. The flag adds the appropriate headers when you use `HTTP GET` requests to retrieve resources from the Storage account.

https://docs.microsoft.com/en-us/rest/api/storageservices/cross-origin-resource-sharing--cors--support-for-the-azure-storage-services

Question 5: Skipped

Activities within Azure Data Factory define the actions that will be performed on the data and there are three categories including:

• Data movement activities

• Data transformation activities

• Control activities

Pipelines in Data Factory are defined in JSON format as follows:

```
JSON
{
"name": "PipelineName",
"properties":
{
"description": "pipeline description",
"activities":
[
],
"parameters": {
}
}
}
```

Which of the JSON properties are required? (Select all that apply)

- ☐ parameters
- ☐ activities **(Correct)**
- ☐ name **(Correct)**
- ☐ description

Explanation

Activities within Azure Data Factory define the actions that will be performed on the data and there are three categories including:

• Data movement activities

• Data transformation activities

• Control activities

Activities and pipelines

Defining pipelines

Here is how a pipeline is defined in JSON format:

```
JSON
{
"name": "PipelineName",
"properties":
{
"description": "pipeline description",
"activities":
[
],
"parameters": {
}
}
}
```

The following describes properties in the above JSON:

Property: name

Name of the activity.

Required: Yes

Property: description

Text describing what the pipeline is used for.

Required: No

Property: activities

The activities section can have one or more activities defined within it..

Required: Yes

Property: parameters

The parameters section can have one or more parameters defined within the pipeline, making your pipeline flexible for reuse.

Required: No

https://docs.microsoft.com/en-us/azure/data-factory/concepts-pipelines-activities

Question 6: Skipped

Init Scripts provide a way to configure cluster's nodes. It is recommended to favour Cluster Scoped Init Scripts over Global and Named scripts.

Which of the following is best described by:

"You specify the script in cluster's configuration by either writing it directly in the cluster configuration UI or storing it on DBFS and specifying the path in Cluster Create API. Any location under `DBFS /databricks` *folder except* `/databricks/init` *can be used for this purpose."*

- Interactive
- Global
- Cluster Scoped
  (Correct)
- Cluster Named

Explanation
Favour cluster scoped init scripts over global and named scripts

Init Scripts provide a way to configure cluster's nodes and to perform custom installs. Init scripts can be used in the following modes:

• **Global:** by placing the Init script in `/databricks/init` folder, you force the script's execution every time any cluster is created or restarted by users of the workspace.

• **Cluster Named (deprecated):** you can limit the init script to run only on for a specific cluster's creation and restarts by placing it in `/databricks/init/<cluster_name>` folder.

• **Cluster Scoped**: in this mode, the Init script is not tied to any cluster by its name and its automatic execution is not a virtue of its dbfs location. Rather, you specify the script in cluster's configuration by either writing it directly in the cluster configuration UI or storing it on Databricks File System (DBFS) and specifying the path in Cluster Create API. Any location under `DBFS /databricks` folder except `/databricks/init` can be used for this purpose, such as: `/databricks/<my-directory>/set-env-var.sh`

You should treat Init scripts with *extreme* caution because they can easily lead to intractable cluster launch failures. If you really need them, please use the **Cluster Scoped execution mode** as much as possible because:

• ADB executes the script's body in each cluster node. Thus, a successful cluster launch and subsequent operation are predicated on all nodal Init scripts executing in a timely manner without any errors and reporting a zero exit code. This process is highly error prone, especially for scripts downloading artifacts from an external service over unreliable and/or misconfigured networks.

• Because Global and Cluster Named Init scripts execute automatically due to their placement in a special DBFS location, it is easy to overlook that they could be causing a cluster to not launch. By specifying the Init script in the Configuration, there's a higher chance that you'll consider them while debugging launch failures.

Use cluster log delivery feature to manage logs

By default, Cluster logs are sent to default DBFS but you should consider sending the logs to a blob store location under your control using the Cluster Log Delivery feature. The Cluster Logs contain logs emitted by user code, as well as Spark framework's Driver and Executor logs. Sending them to a blob store controlled by yourself is recommended over default DBFS location because:

• ADB's automatic 30-day default DBFS log purging policy might be too short for certain compliance scenarios. A blob store location in your subscription will be free from such policies.

• You can ship logs to other tools only if they are present in your storage account and a resource group governed by you. The root DBFS, although present in your subscription, is launched inside a Microsoft Azure managed resource group and is protected by a read lock. Because of this lock, the logs are only accessible by privileged Azure Databricks framework code. However, constructing a pipeline to ship the logs to downstream log analytics tools requires logs to be in a lock-free location first.

https://github.com/Azure/AzureDatabricksBestPractices/blob/master/toc.md

Question 7: Skipped

Which workload management feature influences the order in which a request gets access to resources?

- Workload importance
  (Correct)
- Workload priority
- Workload isolation
- Workload classification

Explanation
Workload importance indexes write multiple data types and values per row of data and so compression functions are less likely to be able to reduce the size through pattern matching and offsets.

Workload importance

Workload importance influences the order in which a request gets access to resources. On a busy system, a request with higher importance has first access to resources. Importance can also ensure ordered access to locks. There are five levels of importance: low, below_normal, normal, above_normal, and high. Requests that don't set importance are assigned the default level of normal. Requests that have the same importance level have the same scheduling behaviour that exists today.

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/sql-data-warehouse-workload-management

Question 8: Skipped

Authentication is the process of validating credentials as you access resources in a digital infrastructure. This ensures that you can validate that an individual, or a service that wants to access a service in your environment can prove who they are. Azure Synapse Analytics provides several different methods for authentication.

Which are valid authentication methods in Azure Synapse Analytics? (Select all that apply)

- SAML

- ☐ Azure Key Vault
  (Correct)
- ☐ OAuth
- ☐ SQL Authentication
  (Correct)
- ☐ Azure Active Directory
  (Correct)
- ☐ SAS
  (Correct)
- ☐ SSL
- ☐ MFA
  (Correct)
- ☐ Managed identity
  (Correct)

Explanation
Authentication is the process of validating credentials as you access resources in a digital infrastructure. This ensures that you can validate that an individual, or a service that wants to access a service in your environment can prove who they are. Azure Synapse Analytics provides several different methods for authentication.

What needs to be authenticated

There are a variety of scenarios that means that authentication must take place to protect the data that is stored in your Azure Synapse Analytics estate.

The common form of authentication is that of individuals who want to access the data in the service. This is typically seen as an individual providing a username and password to authenticate against a service. However, this is also becoming more sophisticated with authentication requests working in combination with conditional access policies to further secure the authentication process with additional security steps.

What is less obvious is the fact that services must authenticate with other services so that they can operate seamlessly. An example of this is using an Azure Synapse Spark or serverless SQL pool to access data in an Azure Data Lake store. An authentication mechanism must take place in the background to ensure that Azure Synapse Analytics can access the data in the data lake in an authenticated manner.

Finally, there are situations where users and services operate together at the same time. Here you have a combination of both user and service authentication taking place under the hood to ensure that the user is getting access to the data seamlessly. An example of this is using Power BI to view reports in a dashboard that is being serviced by a dedicated SQL pool. Here you have multiple levels of authentication taking place that needs to be managed.

Types of security

The following are the types of authentication that you should be aware of when working with Azure Synapse Analytics.

Azure Active Directory

Azure Active Directory is a directory service that allows you to centrally maintain objects that can be secured. The objects can include user accounts and computer accounts. An employee of an organization will typically have a user account that represents them in the organizations Azure Active Directory tenant, and they then use the user account with a password to authenticate against other resources that are stored within the directory using a process known as single sign-on.

The power of Azure Active Directory is that they only have to login once, and Azure Active Directory will manage access to other resources based on the information held within it using pass through authentication. If a user and an instance of Azure Synapse Analytics are part of the same Azure Active Directory, it is possible for the user to access Azure Synapse Analytics without an apparent login. If managed correctly, this process is seamless as the administrator would have given the user authorization to access Azure Synapse Analytics dedicated SQL pool as an example.

In this situation, it is normal for an Azure Administrator to create the user accounts and assign them to the appropriate roles and groups in Azure Active Directory. The Data Engineer will then add the user, or a group to which the user belongs to access a dedicated SQL pool.

Managed identities

Managed identity for Azure resources is a feature of Azure Active Directory. The feature provides Azure services with an automatically managed identity in Azure AD. You can

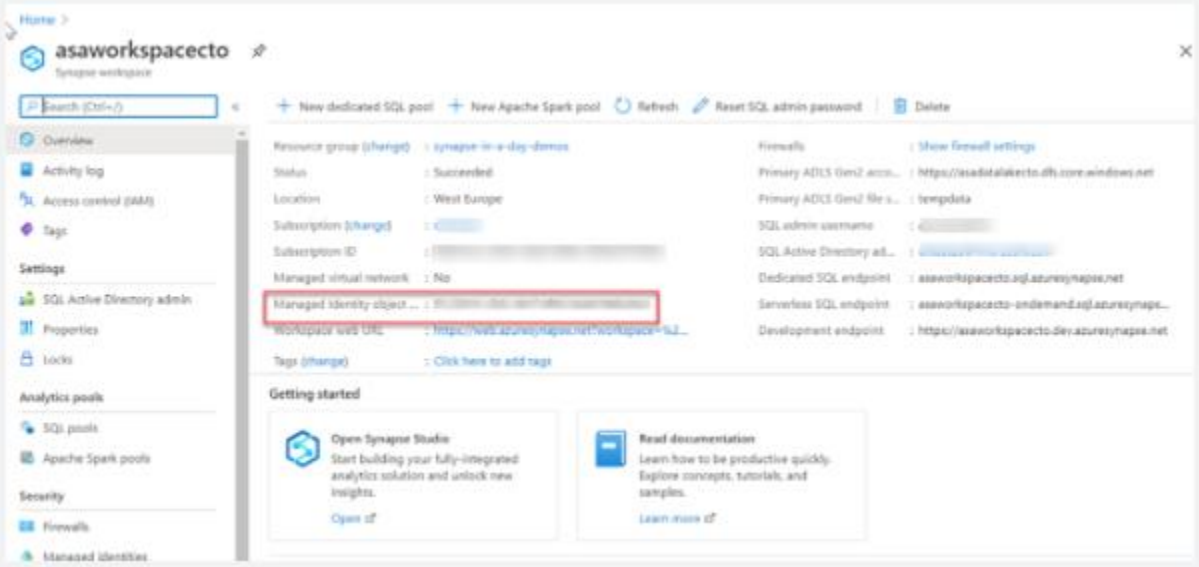
use the Managed Identity capability to authenticate to any service that support Azure Active Directory authentication.

Managed identities for Azure resources are the new name for the service formerly known as Managed Service Identity (MSI). A system-assigned managed identity is created for your Azure Synapse workspace when you create the workspace.

Azure Synapse also uses the managed identity to integrate pipelines. The managed identity lifecycle is directly tied to the Azure Synapse workspace. If you delete the Azure Synapse workspace, then the managed identity is also cleaned up.

The workspace managed identity needs permissions to perform operations in the pipelines. You can use the object ID or your Azure Synapse workspace name to find the managed identity when granting permissions.

You can retrieve the managed identity in the Azure portal. Open your Azure Synapse workspace in Azure portal and select **Overview** from the left navigation. The managed identity's object ID is displayed to in the main screen.



The managed identity information will also show up when you create a linked service that supports managed identity authentication from Azure Synapse Studio.

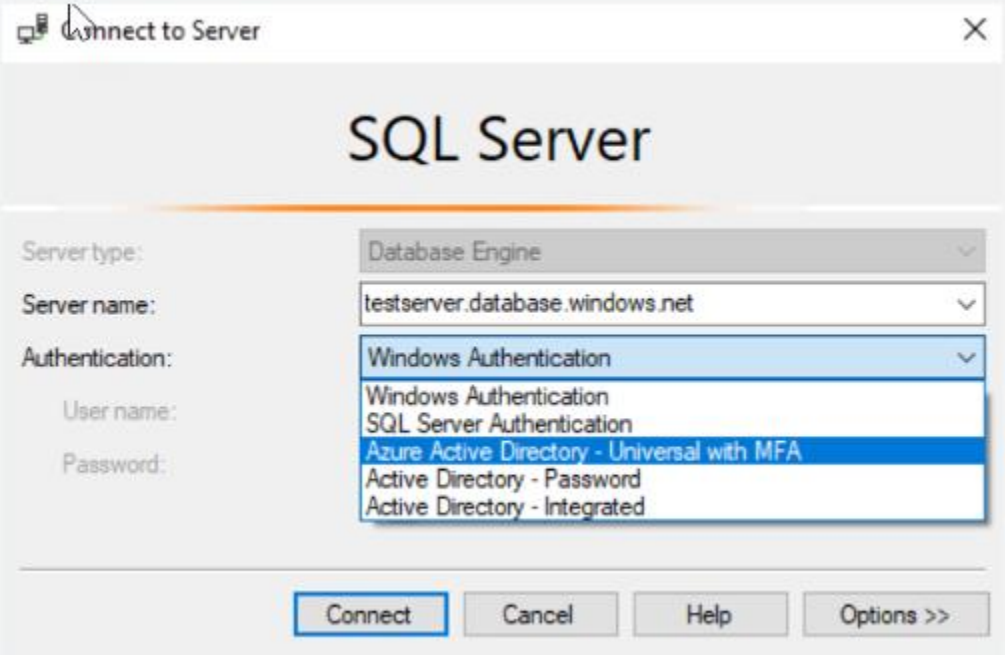
SQL Authentication

For user accounts that are not part of an Azure Active directory, then using SQL Authentication will be an alternative. In this instance, a user is created in the instance of a dedicated SQL pool. If the user in question requires administrator access, then the details of the user are held in the master database. If administrator access is not required, you can create a user in a specific database. A user then connects directly to the Azure Synapse Analytics dedicated SQL pool where they are prompted to use a username and password to access the service.

This approach is typically useful for external users who need to access the data, or if you are using third party or legacy applications against the Azure Synapse Analytics dedicated SQL pool.

Multi factor authentication

Synapse SQL support connections from SQL Server Management Studio (SSMS) using Active Directory Universal Authentication.



This enables you to operate in environments that use conditional access policies that enforce multi-factor authentication as part of the policy.

Keys

If you are unable to use a managed identity to access resources such as Azure Data Lake then you can use storage account keys and shared access signatures.

With storage account keys. Azure creates two of these keys (primary and secondary) for each storage account you create. The keys give access to everything in the account. You'll find the storage account keys in the Azure portal view of the storage account. Just select **Settings**, and then click **Access keys**.

As a best practice, you shouldn't share storage account keys, and you can use Azure Key Vault to manage and secure the keys.

Azure Key Vault is a secret store: a centralized cloud service for storing app secrets - configuration values like passwords and connection strings that must remain secure at all times. Key Vault helps you control your apps' secrets by keeping them in a single central location and providing secure access, permissions control, and access logging.

The main benefits of using Key Vault are:

• Separation of sensitive app information from other configuration and code, reducing risk of accidental leaks

• Restricted secret access with access policies tailored to the apps and individuals that need them

• Centralized secret storage, allowing required changes to happen in only one place

• Access logging and monitoring to help you understand how and when secrets are accessed

Secrets are stored in individual vaults, which are Azure resources used to group secrets together. Secret access and vault management is accomplished via a REST API, which is also supported by all of the Azure management tools as well as client libraries available for many popular languages. Every vault has a unique URL where its API is hosted.

Shared access signatures

If an external third-party application need access to your data, you'll need to secure their connections without using storage account keys. For untrusted clients, use a shared access signature (SAS). A shared access signature is a string that contains a security token that can be attached to a URI. Use a shared access signature to delegate access to storage objects and specify constraints, such as the permissions and the time range of access. You can give a customer a shared access signature token.

Types of shared access signatures

You can use a service-level shared access signature to allow access to specific resources in a storage account. You'd use this type of shared access signature, for example, to allow an app to retrieve a list of files in a file system or to download a file.

Use an account-level shared access signature to allow access to anything that a service-level shared access signature can allow, plus additional resources and abilities. For example, you can use an account-level shared access signature to allow the ability to create file systems.

https://docs.microsoft.com/en-us/azure/synapse-analytics/security-baseline

Question 9: Skipped

As great as data lakes are at inexpensively storing our raw data, they also bring with them performance challenges:

• **Too many small or very big files** - more time opening & closing files rather than reading contents (worse with streaming).

• **Partitioning also known as "poor man's indexing"**- breaks down if you picked the wrong fields or when data has many dimensions, high cardinality columns.

• **No caching** - cloud storage throughput is low (cloud object storage is 20-50MB/s/core vs 300MB/s/core for local SSDs).

As a solution to the challenges with Data Lakes noted above, Delta Lake is a file format that can help you build a data lake comprised of one or many tables in Delta Lake format. Delta Lake integrates tightly with Apache Spark, and uses an open format that is based on Parquet.

Two of the core features of Delta Lake are performing `UPSERTS` and Time Travel operations.

What does the Time Travel operation do? (Select all that apply)

- ☐ Writing complex temporal queries.
  (Correct)
- ☐ Providing snapshot isolation for a set of queries for fast changing tables.
  (Correct)
- ☐

Because Delta Lake is version controlled, you have the option to query past versions of the data using a single file storage system.
(Correct)

- Re-creating analyses, reports, or outputs (for example, the output of a machine learning model). This could be useful for debugging or auditing, especially in regulated industries.
(Correct)

Explanation
Delta Lake is a transactional storage layer designed specifically to work with Apache Spark and Databricks File System (DBFS). At the core of Delta Lake is an optimized Spark table. It stores your data as Apache Parquet files in DBFS and maintains a transaction log that efficiently tracks changes to the table.

Data lakes

A data lake is a storage repository that inexpensively stores a vast amount of raw data, both current and historical, in native formats such as `XML`, `JSON`, `CSV`, and `Parquet`. It may contain operational relational databases with live transactional data.

Enterprises have been spending millions of dollars getting data into data lakes with Apache Spark. The aspiration is to do data science and ML on all that data using Apache Spark.

But the data is not ready for data science & ML. The majority of these projects are failing due to unreliable data!

The challenge with data lakes

Why are these projects struggling with reliability and performance?

To extract meaningful information from a data lake, you must solve problems such as:

• Schema enforcement when new tables are introduced.

• Table repairs when any new data is inserted into the data lake.

• Frequent refreshes of metadata.

• Bottlenecks of small file sizes for distributed computations.

• Difficulty sorting data by an index if data is spread across many files and partitioned.

There are also data reliability challenges with data lakes:

• Failed production jobs leave data in corrupt state requiring tedious recovery.

• Lack of schema enforcement creates inconsistent and low quality data.

• Lack of consistency makes it almost impossible to mix appends and reads, batch and streaming.

As great as data lakes are at inexpensively storing our raw data, they also bring with them performance challenges:

• **Too many small or very big files** - more time opening & closing files rather than reading contents (worse with streaming).

• **Partitioning also known as "poor man's indexing"**- breaks down if you picked the wrong fields or when data has many dimensions, high cardinality columns.

• **No caching** - cloud storage throughput is low (cloud object storage is 20-50MB/s/core vs 300MB/s/core for local SSDs).

The solution: Delta Lake

Delta Lake is a file format that can help you build a data lake comprised of one or many tables in Delta Lake format. Delta Lake integrates tightly with Apache Spark, and uses an open format that is based on Parquet. Because it is an open-source format, Delta Lake is also supported by other data platforms, including Azure Synapse Analytics.

Delta Lake makes data ready for analytics.

Delta Lake is an open-source storage layer that brings ACID transactions to Apache Spark™ and big data workloads.

You can read and write data that's stored in Delta Lake by using Apache Spark SQL batch and streaming APIs. These are the same familiar APIs that you use to work with Hive tables or DBFS directories. Delta Lake provides the following functionality:

ACID Transactions: Data lakes typically have multiple data pipelines reading and writing data concurrently, and data engineers have to go through a tedious process to ensure data integrity, due to the lack of transactions. Delta Lake brings ACID transactions to your data lakes. It provides serializability, the strongest level of isolation level.

Scalable Metadata Handling: In big data, even the metadata itself can be "big data". Delta Lake treats metadata just like data, leveraging Spark's distributed processing power to handle all its metadata. As a result, Delta Lake can handle petabyte-scale tables with billions of partitions and files at ease.

Time Travel (data versioning): Delta Lake provides snapshots of data enabling developers to access and revert to earlier versions of data for audits, rollbacks or to reproduce experiments.

Open Format: All data in Delta Lake is stored in Apache Parquet format enabling Delta Lake to leverage the efficient compression and encoding schemes that are native to Parquet.

Unified Batch and Streaming Source and Sync: A table in Delta Lake is both a batch table, as well as a streaming source and sync. Streaming data ingest, batch historic backfill, and interactive queries all just work out of the box.

Schema Enforcement: Delta Lake provides the ability to specify your schema and enforce it. This helps ensure that the data types are correct and required columns are present, preventing bad data from causing data corruption.

Schema Evolution: Big data is continuously changing. Delta Lake enables you to make changes to a table schema that can be applied automatically, without the need for cumbersome DDL.

100% Compatible with Apache Spark API: Developers can use Delta Lake with their existing data pipelines with minimal change as it is fully compatible with Spark, the commonly used big data processing engine.

Get started with Delta using Spark APIs

Delta Lake is included with Azure Databricks. You can start using it today. To quickly get started with Delta Lake, do the following:

Instead of parquet...

```
Python
CREATE TABLE ...
USING parquet
...


dataframe
.write
.format("parquet")
.save("/data")
... simply say delta
```

```
Python
CREATE TABLE ...
USING delta
...

dataframe
.write
.format("delta")
.save("/data")
```

Using Delta with your existing Parquet tables

Step 1: Convert Parquet to Delta tables:

```
Python
CONVERT TO DELTA parquet.`path/to/table` [NO STATISTICS]
[PARTITIONED BY (col_name1 col_type1, col_name2 col_type2, ...)]
Step 2: Optimize layout for fast queries:
Python
OPTIMIZE events
WHERE date >= current_timestamp() - INTERVAL 1 day
ZORDER BY (eventType)
```

Basic syntax

Two of the core features of Delta Lake are performing upserts (insert/updates) and Time Travel operations.

To `UPSERT` means to "UPdate" and "inSERT". In other words, `UPSERT` is literally TWO operations. It is not supported in traditional data lakes, as running an UPDATE could invalidate data that is accessed by the subsequent INSERT operation.

Using Delta Lake, however, we can do `UPSERTS`. Delta Lake combines these operations to guarantee atomicity to

• `INSERT` a row

• if the row already exists, `UPDATE` the row.

Upsert syntax

Upserting, or merging, in Delta Lake provides fine-grained updates of your data. The following syntax shows how to perform an Upsert:

```
SQL
MERGE INTO customers -- Delta table
USING updates
ON customers.customerId = source.customerId
WHEN MATCHED THEN
UPDATE SET address = updates.address
WHEN NOT MATCHED
THEN INSERT (customerId, address) VALUES (updates.customerId, updates.address)
```

Time Travel syntax

Because Delta Lake is version controlled, you have the option to query past versions of the data. Using a single file storage system, you now have access to several versions your historical data, ensuring that your data analysts will be able to replicate their reports (and compare aggregate changes over time) and your data scientists will be able to replicate their experiments.

Other time travel use cases are:

• Re-creating analyses, reports, or outputs (for example, the output of a machine learning model). This could be useful for debugging or auditing, especially in regulated industries.

• Writing complex temporal queries.

• Fixing mistakes in your data.

• Providing snapshot isolation for a set of queries for fast changing tables.

Example of using time travel to reproduce experiments and reports:

```
SQL
SELECT count(*) FROM events
TIMESTAMP AS OF timestamp


SELECT count(*) FROM events
```

```
VERSION AS OF version

Python

spark.read.format("delta").option("timestampAsOf", timestamp_string).load("/event
s/")

If you need to rollback accidental or bad writes:

SQL

INSERT INTO my_table

SELECT * FROM my_table TIMESTAMP AS OF

date_sub( current_date(), 1)
```

https://docs.microsoft.com/en-us/azure/synapse-analytics/spark/apache-spark-what-is-delta-lake

Question 10: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

A(n) [?] schema may be defined at query time.

- Unstructured data type
  (Correct)
- Structured data type
- Azure Cosmos DB data type
- Hybrid data type

Explanation
An Unstructured data type schema may be defined at query time.

The schema of unstructured data is typically defined at query time. This means that data can be loaded onto a data platform in its native format.

https://docs.microsoft.com/en-us/azure/architecture/guide/technology-choices/data-store-overview

Question 11: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

In Azure Data Factory, a(n) [?] is a logical grouping of activities that together perform a task.

- Sink
- Orchestration
- Linked Service
- Pipeline
  (Correct)
- Activity

Explanation
A pipeline in Azure Data Factory is a logical grouping of activities such as copy in order to perform a task. The activity defines the operation that you're performing on the data (therefore, a copy means copying the same data to another data store). For example, a pipeline could contain a set of activities that ingest and clean log data, and then kick off a mapping data flow to analyze the log data.

https://docs.microsoft.com/en-us/azure/data-factory/concepts-pipelines-activities

Question 12: Skipped

What does Azure Data Lake Storage (ADLS) Passthrough enable?

- Blocking ADLS resources through a mount point when credential passthrough is enabled.
- Automatically mounting ADLS accounts to the workspace that are added to the managed resource group.
- Commands running on a configured cluster can read and write data in ADLS without configuring service principal credentials.
  (Correct)
- User security groups that are added to ADLS are automatically created in the workspace as Databricks groups.

Explanation
Azure Data Lake Storage (ADLS) Passthrough enables commands running on a configured cluster can read and write data in ADLS without configuring service principal

credentials. In addition, authentication to ADLS from Azure Databricks clusters is automatic, using the same Azure AD identity one uses to log into Azure Databricks.

https://docs.microsoft.com/en-us/azure/databricks/security/credential-passthrough/adls-passthrough

Question 13: Skipped

Azure provides many ways to store your data and there are several tools that create a storage account.

Which aspects guide a user's decision on the tool used to create a storage account? (Select two)

- ☐ The datatype being stored in the account
- ☐ Tool cost
- ☐ If the user needs automation **(Correct)**
- ☐ Location restrictions of the data centre
- ☐ If the user wants a GUI **(Correct)**

Explanation
There are several tools that create a storage account. Your choice is typically based on if you want a GUI and whether you need automation.

Available tools

The available tools are:

• Azure Portal

• Azure CLI (Command-line interface)

• Azure PowerShell

• Management client libraries

The portal provides a GUI with explanations for each setting. This makes the portal easy to use and helpful for learning about the options.

The other tools in the above list all support automation. The Azure CLI and Azure PowerShell let you write scripts, while the management libraries allow you to incorporate the creation into a client app.

How to choose a tool

Storage accounts are typically based on an analysis of your data, so they tend to be relatively stable. As a result, storage-account creation is usually a one-time operation done at the start of a project. For one-time activities, the portal is the most common choice.

In the rare cases where you need automation, the decision is between a programmatic API or a scripting solution. Scripts are typically faster to create and less work to maintain because there is no need for an IDE, NuGet packages, or build steps. If you have an existing client application, the management libraries might be an attractive choice; otherwise, scripts will likely be a better option.

https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

Question 14: Skipped

When doing a write stream command, what does the `outputMode("append")` option do?

- The append `outputMode` allows records to update to the output log.
- The append mode allows records to be updated and changed in place.
- The append mode replaces existing records and updates aggregates.
- The append `outputMode` allows records to be added to the output sink. **(Correct)**

Explanation

The `outputMode` "append" option informs the write stream to add only new records to the output sink. The "complete" option is to rewrite the full output - applicable to aggregations operations. Finally, the "update" option is for updating changed records in place.

https://jaceklaskowski.gitbooks.io/spark-structured-streaming/content/spark-sql-streaming-MemorySink.html

Question 15: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

[A] data is information that doesn't reside in a relational database but still has some structure to it. Databases that hold documents are held in [B] format.

- [A] Structured, [B] Relational
- [A] Semi-Structured, [B] JSON
  (Correct)
- [A] JSON, [B] Semi-Structured
- [A] Unstructured, [B] Binary

Explanation

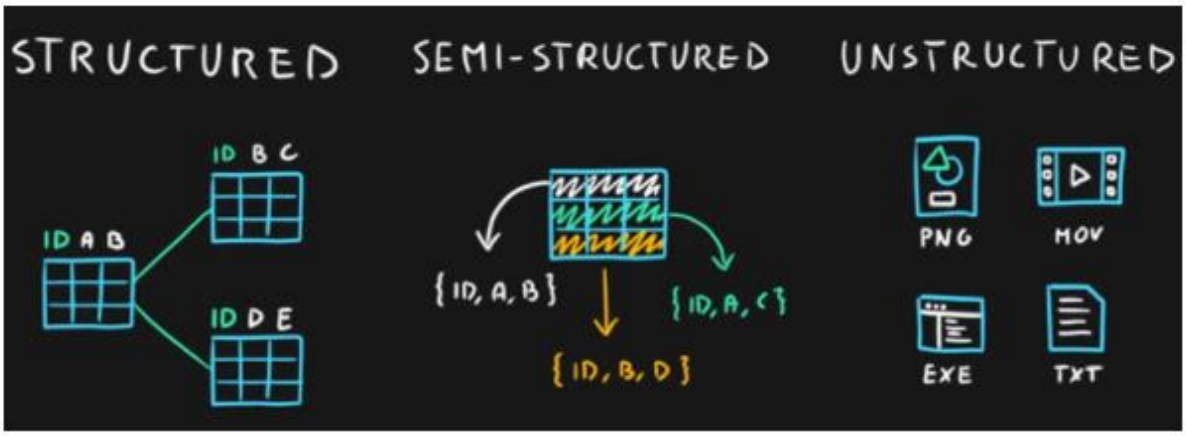
Semi-structured data is information that doesn't reside in a relational database but still has some structure to it. Examples include documents held in *JavaScript Object Notation* (JSON) format. The example below shows a pair of documents representing customer information.

```
## Document 1 ##
{
  "customerID": "103248",
  "name":
  {
    "first": "AAA",
    "last": "BBB"
  },
  "address":
  {
    "street": "Main Street",
    "number": "101",
    "city": "Acity",
    "state": "NY"
  },
  "ccOnFile": "yes",
  "firstOrder": "02/28/2003"
}
```

There are other types of semi-structured data as well. Examples include *key-value* stores and *graph* databases.

A key-value store is similar to a relational table, except that each row can have any number of columns.

You can use a graph database to store and query information about complex relationships. A graph contains nodes (information about objects), and edges (information about the relationships between objects). The image below shows an example of how you might structure the data in a graph database.



https://f5a395285c.nxcli.net/microsoft-azure/dp-900/structured-data-vs-unstructured-data-vs-semi-structured-data/

Question 16: Skipped

Which tool is used to perform an assessment of migrating SSIS packages to Azure SQL Database services?

- Lab Services
- Data Migration Assessment
- Data Migration Assistant
  (Correct)
- SQL Server Management Studio
- ARM templates
-

- Data Migration Service
- SQL Server Upgrade Advisor

Explanation
The Data Migration Assistant is used to perform an assessment of migrating SSIS packages to Azure SQL Database services.

To assess SQL Server Integration Service(SSIS) packages, below components need to be installed with Data Migration Assistant:

• SQL Server Integration Service with the same version as the SSIS packages to assess.

• Azure Feature Pack or other third party components if SSIS packages to assess have these components.

DMA needs to run with **administrator** access to assess SSIS packages in Package Store.

https://docs.microsoft.com/en-us/sql/dma/dma-assess-ssis?view=sql-server-ver15

Question 17: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

Many business application architectures separate transactional and analytical processing into separate systems with data stored and processed on separate infrastructures. These infrastructures are commonly referred to as OLTP (online transaction processing) systems working with operational data, and OLAP (online analytical processing) systems working with historical data, with each system is optimized for their specific task.

Azure Cosmos DB provides … [?]

- None of the listed options.
- A transactional store optimized for transactional workloads and a fully managed autosync process to keep the data within these stores in sync.
- An analytical store optimized for analytical workloads and a fully managed autosync process to keep the data within these stores in sync.
-

Both a transactional store optimized for transactional workloads and an analytical store optimized for analytical workloads and a fully managed autosync process to keep the data within these stores in sync.
(Correct)

Explanation
Many business application architectures separate transactional and analytical processing into separate systems with data stored and processed on separate infrastructures. These infrastructures are commonly referred to as OLTP (online transaction processing) systems working with operational data, and OLAP (online analytical processing) systems working with historical data, with each system is optimized for their specific task.

OLTP systems are optimized for dealing with discrete system or user requests immediately and responding as quickly as possible.

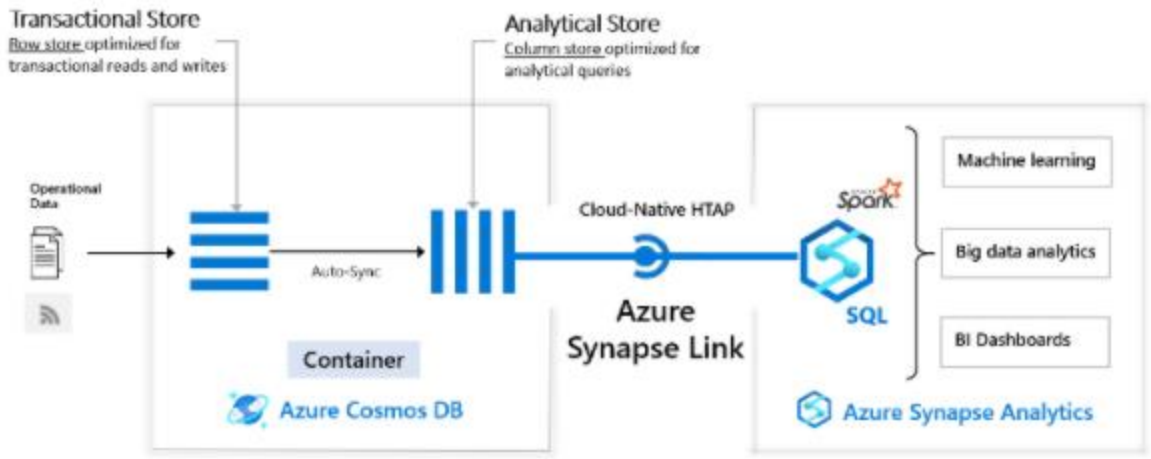
OLAP systems are optimized for the analytical processing, ingesting, synthesizing, and managing large sets of historical data. The data processed by OLAP systems largely originates from OLTP systems and needs to be loaded into the OLTP systems by means of batch processes commonly referred to as ETL (Extract, Transform, and Load) jobs.

Due to their complexity and the need to physically copy large amounts of data, this creates a delay in data being available to provide insights by way of the OLAP systems.

As more and more businesses move to digital processes, they increasingly recognize the value of being able to respond to opportunities by making faster and well-informed decisions. HTAP (Hybrid Transactional/Analytical processing) enables business to run advanced analytics in near-real-time on data stored and processed by OLTP systems.

Azure Synapse Link for Azure Cosmos DB

Azure Synapse Link for Azure Cosmos DB is a cloud-native HTAP capability that enables you to run near-real-time analytics over operational data stored in Azure Cosmos DB. Azure Synapse Link creates a tight seamless integration between Azure Cosmos DB and Azure Synapse Analytics.



Azure Cosmos DB provides both a transactional store optimized for transactional workloads and an analytical store optimized for analytical workloads and a fully managed autosync process to keep the data within these stores in sync.

Azure Synapse Analytics provides both a SQL Serverless query engine for querying the analytical store using familiar T-SQL and an Apache Spark query engine for leveraging the analytical store using your choice of Scala, Java, Python or SQL and provides a user-friendly notebook experience.

Together Azure Cosmos DB and Synapse Analytics enable organizations to generate and consume insights from their operational data in near-real time, using the query and analytics tools of their choice. All of this is achieved without the need for complex ETL pipelines and without affecting the performance of their OLTP systems using Azure Cosmos DB.

https://docs.microsoft.com/en-us/azure/cosmos-db/synapse-link

Question 18: Skipped

Scenario: A customer of Ultron Electronics is attempting to use a $300 store credit for the full amount of a new purchase. They are trying to double-spend their credit by creating two transactions at the exact same time using the entire store credit. The customer is making two transactions using two different devices.

The database behind the scenes is an ACID-compliant transactional database.

What would be the result?

- None of the listed options.
- Both orders would be processed and use the in-store credit.
- One order would be processed and use the in-store credit, and the other order would update the remaining inventory for the items in the basket, but would not complete the order.
- One order would be processed and use the in-store credit, and the other order would not be processed.
  (Correct)

Explanation
Once the second order determined that the in-store credit has already been used, it would roll back the transaction.

A transactional database must adhere to the **ACID (Atomicity, Consistency, Isolation, Durability)** properties to ensure that the database remains consistent while processing transactions.

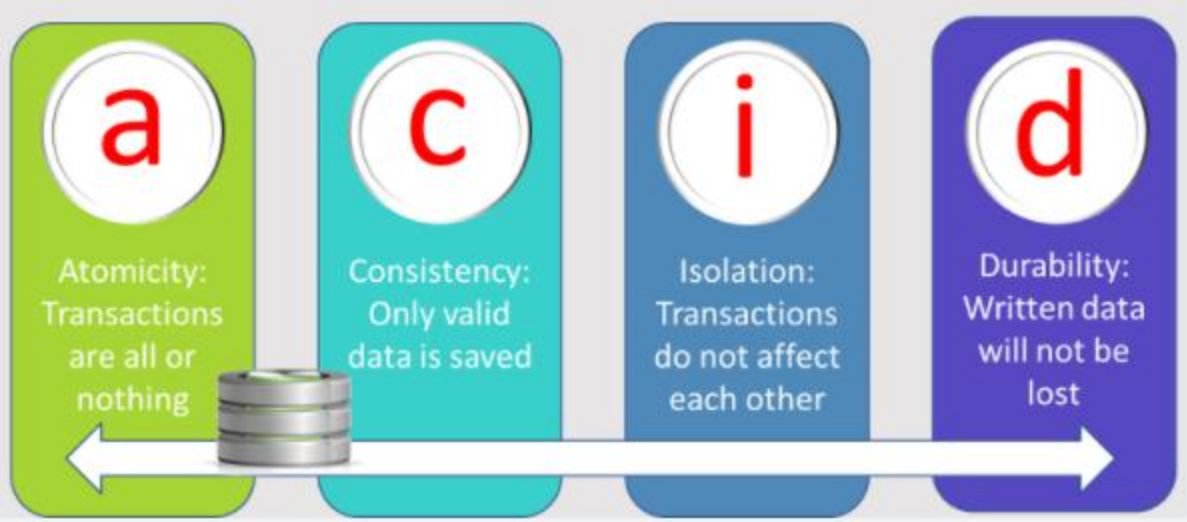
The four letters in ACID represent the four required characteristics of database transactions:

• Atomicity

• Consistency

• Isolation

• Durability



• *Atomicity* guarantees that each transaction is treated as a single *unit*, which either succeeds completely, or fails completely. If any of the statements constituting a transaction fails to complete, the entire transaction fails and the database is left unchanged. An atomic system must guarantee atomicity in each and every situation, including power failures, errors, and crashes.

• *Consistency* ensures that a transaction can only take the data in the database from one valid state to another. A consistent database should never *lose* or *create* data in a manner that can't be accounted for. In the bank transfer example described earlier, if you add funds to an account, there must be a corresponding deduction of funds somewhere, or a record that describes where the funds have come from if they have been received externally. You can't suddenly create (or lose) money.

• *Isolation* ensures that concurrent execution of transactions leaves the database in the same state that would have been obtained if the transactions were executed sequentially. A concurrent process can't see the data in an inconsistent state (for example, the funds have been deducted from one account, but not yet credited to another.)

• *Durability* guarantees that once a transaction has been committed, it will remain committed even if there's a system failure such as a power outage or crash.

https://www.techopedia.com/definition/23949/atomicity-consistency-isolation-durability-acid-database-management-system

Question 19: Skipped

Scenario: You are working as a consultant at **Advanced Idea Mechanics** (**A.I.M.**) who is a privately funded think tank organized of a group of brilliant scientists whose sole dedication is to acquire and develop power through technological means. Their goal is to use this power to overthrow the governments of the world. They supply arms and technology to radicals and subversive organizations in order to foster a violent technological revolution of society while making a profit.

The company has 10,000 employees. Most employees are located in Europe. The company supports teams worldwide.

AIM has two main locations: a main office in London, England, and a manufacturing plant in Berlin, Germany.

AIM plans to implement an Azure Cosmos DB database which will be replicated to four global regions where only the one closest to London will be writable. During events, the Cosmos DB will write 250,000 JSON each day and the consistency level must meet the following.

Requirements:

· The system must guarantee monotonic reads and writes within a session.

· The system must provide the fastest throughput available.

· Latency must be the lowest available.

As the expert, the team looks to you for direction. Which of the following consistency levels should you advise them to utilize?

- Session
  (Correct)
- Strong
- Eventual
- Consistent Prefix
- Bounded Staleness

Explanation
The key phrase is "The system must guarantee monotonic reads and writes within a session."

Consistency levels in Azure Cosmos DB

Distributed databases that rely on replication for high availability, low latency, or both, must make a fundamental tradeoff between the read consistency, availability, latency, and throughput as defined by the PACLC theorem. The linearizability of the strong consistency model is the gold standard of data programmability. But it adds a steep price from higher write latencies due to data having to replicate and commit across large distances. Strong consistency may also suffer from reduced availability (during failures) because data cannot replicate and commit in every region. Eventual consistency offers higher availability and better performance, but its more difficult to program applications because data may not be completely consistent across all regions.

Most commercially available distributed NoSQL databases available in the market today provide only strong and eventual consistency. Azure Cosmos DB offers five well-defined levels. From strongest to weakest, the levels are:

• *Strong*

• *Bounded staleness*

• *Session*

• *Consistent prefix*

• *Eventual*

Each level provides availability and performance tradeoffs. The following image shows the different consistency levels as a spectrum.

The consistency levels are region-agnostic and are guaranteed for all operations regardless of the region from which the reads and writes are served, the number of regions associated with your Azure Cosmos account, or whether your account is configured with a single or multiple write regions.

Session consistency

In session consistency, within a single client session reads are guaranteed to honor the consistent-prefix, monotonic reads, monotonic writes, read-your-writes, and write-follows-reads guarantees. This assumes a single "writer" session or sharing the session token for multiple writers.

• Clients outside of the session performing writes will see the following guarantees:

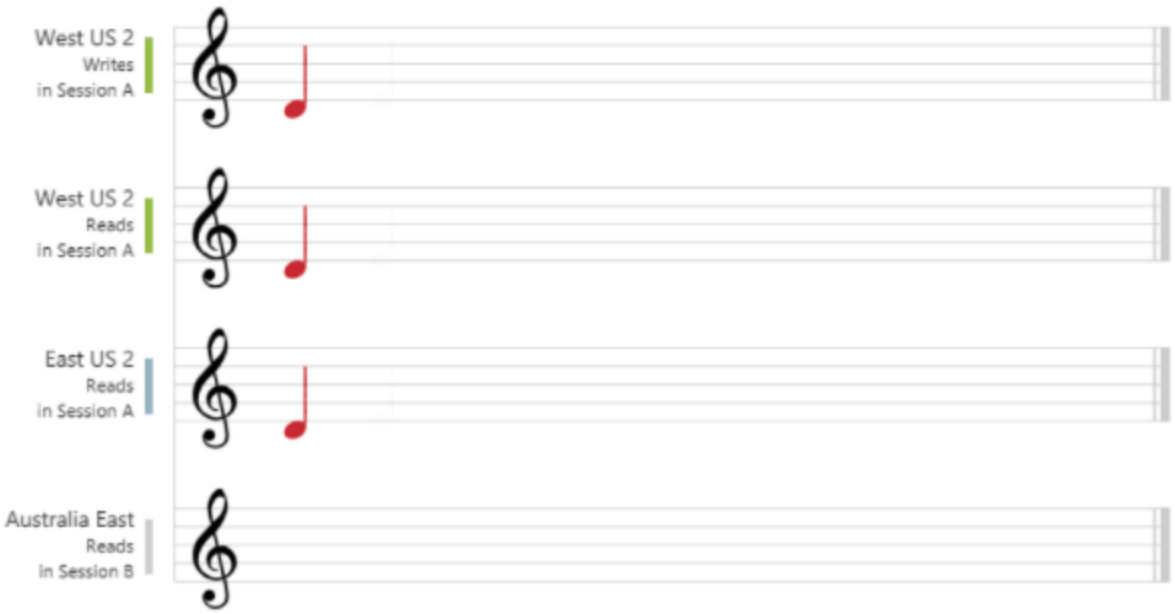
• Consistency for clients in same region for an account with single write region = Consistent Prefix

• Consistency for clients in different regions for an account with single write region = Consistent Prefix

• Consistency for clients writing to a single region for an account with multiple write regions = Consistent Prefix

• Consistency for clients writing to multiple regions for a account with multiple write regions = Eventual

Session consistency is the most widely used consistency level for both single region as well as globally distributed applications. It provides write latencies, availability, and read throughput comparable to that of eventual consistency but also provides the consistency guarantees that suit the needs of applications written to operate in the context of a user. The following graphic illustrates the session consistency with musical notes. The "West US 2 writer" and the "West US 2 reader" are using the same session (Session A) so they both read the same data at the same time. Whereas the "Australia East" region is using "Session B" so, it receives data later but in the same order as the writes.



https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels

Question 20: Skipped

How can you manage the lifecycle of data and define how long it will be retained for in an analytical store?

- Configure the cache to set the time to retain the data in memory.
- Configure the default Time to Live (TTL) property for records stored.
  (Correct)
- Configure the purge duration in a container.
- Configure the deletion duration for records in the transactional store.

Explanation
Configuring the default Time to Live (TTL) property for records stored in an analytical store can manage the lifecycle of data and define how long it will be retained for.

https://help.ns1.com/hc/en-us/articles/360022250193-Best-practices-TTL-configuration

Question 21: Skipped

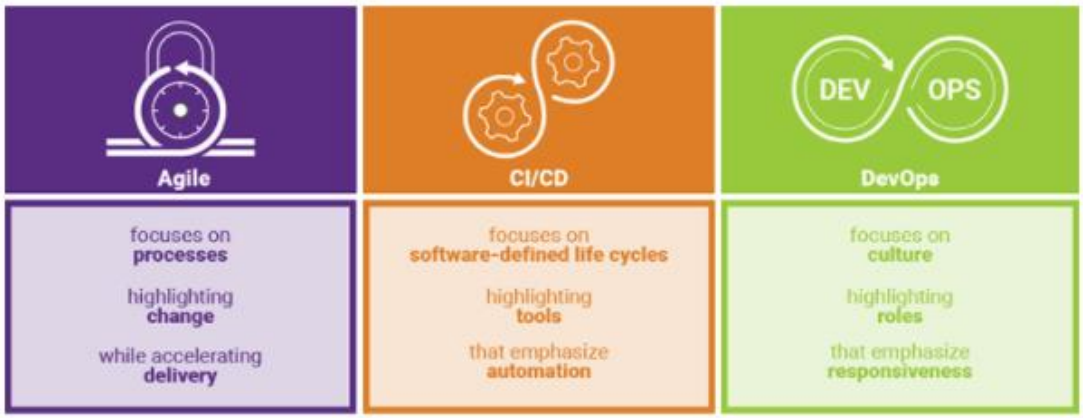
While Agile, CI/CD, and DevOps are different, they support one another

Which is best described by:

“Focuses on software-defined life cycles highlighting tools that emphasize automation.”

- Agile
- DevOps
- CI/CD
  (Correct)
- SDLC

Explanation
While Agile, CI/CD, and DevOps are different, they support one another. Agile focuses on the development process, CI/CD on practices, and DevOps on culture.



• **Agile** focuses on processes highlighting change while accelerating delivery.

• **CI/CD** focuses on software-defined life cycles highlighting tools that emphasize automation.

• **DevOps** focuses on culture highlighting roles that emphasize responsiveness.

https://www.synopsys.com/blogs/software-security/agile-cicd-devops-difference/

Azure DevOps is a collection of services that provide an end-to-end solution for the five core practices of DevOps: planning and tracking, development, build and test, delivery, and monitoring and operations.

It is possible to put an Azure Databricks Notebook under Version Control in an Azure Devops repo. Using Azure DevOps, you can then build Deployment pipelines to manage your release process.

CI/CD with Azure DevOps

Here are some of the features that make it well-suited to CI/CD with Azure Databricks.

• Integrated Git repositories

• Integration with other Azure services

• Automatic virtual machine management for testing builds

• Secure deployment

• Friendly GUI that generates (and accepts) various scripted files

But what is CI/CD?

Continuous Integration

Throughout the development cycle, developers commit code changes locally as they work on new features, bug fixes, etc. If the developers practice continuous integration, they merge their changes back to the main branch as often as possible. Each merge into the master branch triggers a build and automated tests that validate the code changes to ensure successful integration with other incoming changes. This process avoids integration headaches that frequently happen when people wait until the release day before they merge all their changes into the release branch.

Continuous Delivery

Continuous delivery builds on top of continuous integration to ensure you can successfully release new changes in a fast and consistent way. This is because, in addition to the automated builds and testing provided by continuous integration, the release process is automated to the point where you can deploy your application with the click of a button.

Continuous Deployment

Continuous deployment takes continuous delivery a step further by automatically deploying your application without human intervention. This means that merged changes pass through all stages of your production pipeline and, unless any of the tests fail, automatically release to production in a fully automated manner.

Who benefits?

Everyone. Once properly configured, automated testing and deployment can free up your engineering team and enable your data team to push their changes into production. For example:

• Data engineers can easily deploy changes to generate new tables for BI analysts.

• Data scientists can update models being used in production.

• Data analysts can modify scripts being used to generate dashboards.

In short, changes made to a Databricks notebook can be pushed to production with a simple mouse click (and then any amount of oversight that your DevOps team feels is appropriate).

https://docs.microsoft.com/en-us/azure/devops/user-guide/alm-devops-features?view=azure-devops

Question 22: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

The process of consuming data streams, analyzing them, and deriving actionable insights out of them is called [?].

- Event processing
  (Correct)
- Wrangling
- Multiprocessing
- Consumption

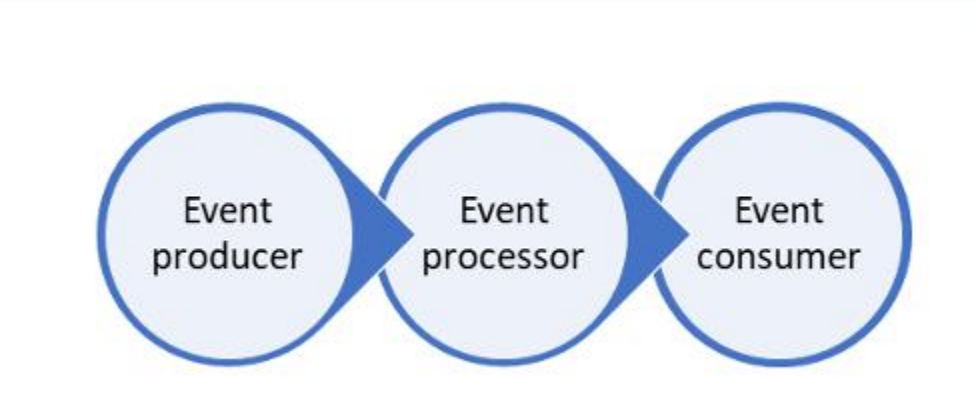
Explanation

The process of consuming data streams, analyzing them, and deriving actionable insights out of them is called event processing. An event processing pipeline has three distinct components:

• **Event producer**: Examples include sensors or processes that generate data continuously, such as a heart rate monitor or a highway toll lane sensor.

• **Event processor**: An engine to consume event data streams and derive insights from them. Depending on the problem space, event processors either process one incoming event at a time, such as a heart rate monitor, or process multiple events at a time, such as Azure Stream Analytics processing the highway toll lane sensor data.

• **Event consumer**: An application that consumes the data and takes specific action based on the insights. Examples of event consumers include alert generation, dashboards, or even sending data to another event processing engine.



https://medium.com/ek-technology/use-azure-iot-solution-build-the-first-stage-of-the-industry-4-0-44e838614f23

Question 23: Skipped

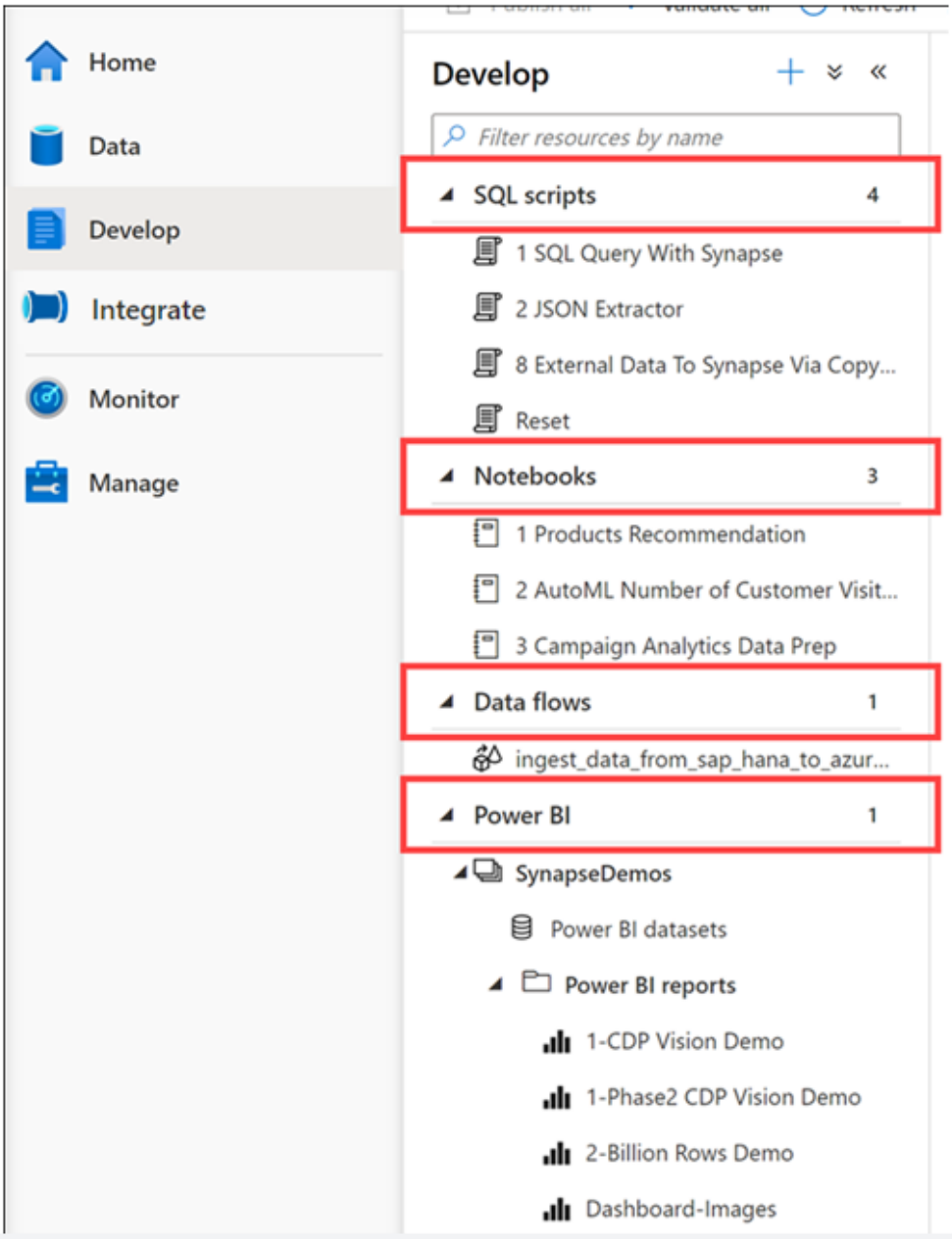
Which Azure Synapse Studio hub would you go to create Notebooks?

- Integrate
- Manage
- None of the listed options **(Correct)**
- Create

- Data

Explanation

In Azure Synapse Studio, the Develop hub is where you manage SQL scripts, Synapse notebooks, data flows, and Power BI reports.



The Develop hub in our sample environment contains examples of the following artifacts:

• **SQL scripts** contains T-SQL scripts that you publish to your workspace. Within the scripts, you can execute commands against any of the provisioned SQL pools or on-demand SQL serverless pools to which you have access.

• **Notebooks** contains Synapse Spark notebooks used for data engineering and data science tasks. When you execute a notebook, you select a Spark pool as its compute target.

• **Data flows** are powerful data transformation workflows that use the power of Apache Spark but are authored using a code-free GUI.

• **Power BI** reports can be embedded here, giving you access to the advanced visualizations they provide without ever leaving the Synapse workspace.

https://www.techtalkcorner.com/azure-synapse-analytics-develop-hub/

Question 24: Skipped

Within the context of Azure Databricks, sharing data from one worker to another can be a costly operation.

Spark has optimized this operation by using a format called [?] which prevents the need for expensive serialization and de-serialization of objects in order to get data from one JVM to another.

- Tungsten
  (Correct)
- Shuffles
- Pipelining
- Stages
- Stage boundary
- Lineage

Explanation
As opposed to narrow transformations, wide transformations cause data to shuffle between executors. This is because a wide transformation requires sharing data across workers. **Pipelining** helps us optimize our operations based on the differences between the two types of transformations.

Pipelining

• Pipelining is the idea of executing as many operations as possible on a single partition of data.

• Once a single partition of data is read into RAM, Spark will combine as many narrow operations as it can into a single **Task**

• Wide operations force a shuffle, conclude a stage, and end a pipeline.

Shuffles

A shuffle operation is triggered when data needs to move between executors.

To carry out the shuffle operation Spark needs to:

• Convert the data to the `UnsafeRow`, commonly referred to as **Tungsten Binary Format**.

• Write that data to disk on the local node - at this point the slot is free for the next task.

• Send that data across the wire to another executor

• Technically the Driver decides which executor gets which piece of data.

• Then the executor pulls the data it needs from the other executor's shuffle files.

• Copy the data back into RAM on the new executor

• The concept, if not the action, is just like the initial read "every“ DataFrame starts with.

• The main difference being it's the 2nd+ stage.

As we will see in a moment, this amounts to a free cache from what is effectively temp files.

Some actions induce in a shuffle. Good examples would include the operations count() and reduce(..).

UnsafeRow (also known as Tungsten Binary Format)

Sharing data from one worker to another can be a costly operation.

Spark has optimized this operation by using a format called **Tungsten**.

Tungsten prevents the need for expensive serialization and de-serialization of objects in order to get data from one JVM to another.

The data that is "shuffled" is in a format known as `UnsafeRow`, or more commonly, the Tungsten Binary Format.

`UnsafeRow` is the in-memory storage format for Spark SQL, DataFrames & Datasets.

Advantages include:

• Compactness:

• Column values are encoded using custom encoders, not as JVM objects (as with RDDs).

• The benefit of using Spark 2.x's custom encoders is that you get almost the same compactness as Java serialization, but significantly faster encoding/decoding speeds.

• Also, for custom data types, it is possible to write custom encoders from scratch.

• Efficiency: Spark can operate *directly out of Tungsten*, without first deserializing Tungsten data into JVM objects.

How UnsafeRow works

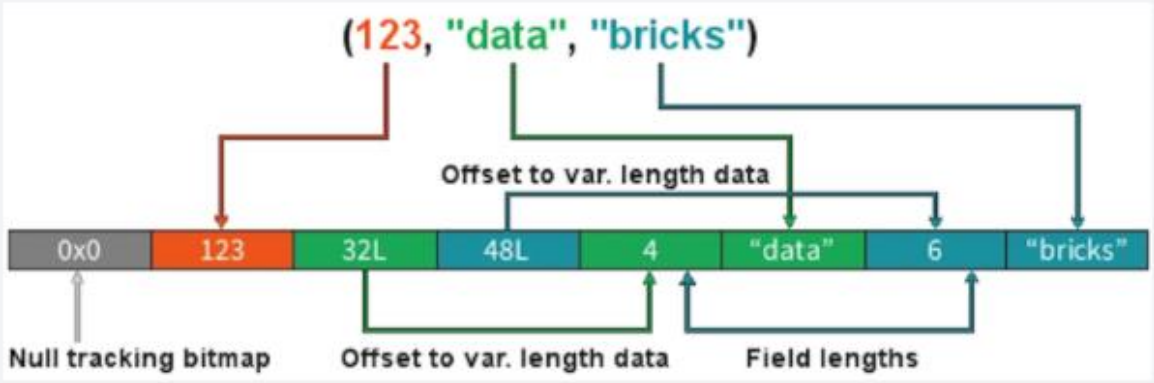
• The first field, "123", is stored in place as its primitive.

• The next 2 fields, "data" and "bricks", are strings and are of variable length.

• An offset for these two strings is stored in place (32L and 48L respectively shown in the picture below).

• The data stored in these two offset's are of format "length + data".

• At offset 32L, we store 4 + "data" and likewise at offset 48L we store 6 + "bricks".



Stages

• When we shuffle data, it creates what is known as a stage boundary.

• Stage boundaries represent a process bottleneck.

Take for example the following transformations:

Step Transformation

1 Read

2 Select

3 Filter

4 GroupBy

5 Select

6 Filter

7 Write

Spark will break this one job into two stages (steps 1-4b and steps 4c-7):

Stage #1

Step Transformation

1 Read

2 Select

3 Filter

4a GroupBy 1/2

4b shuffle write

Stage #1

Step Transformation

4c shuffle read

4d GroupBy 2/2

5 Select

6 Filter

7 Write

In **Stage #1**, Spark will create a pipeline of transformations in which the data is read into RAM (Step #1), and then perform steps #2, #3, #4a & #4b

All partitions must complete **Stage #1** before continuing to **Stage #2.**

• It's not possible to group all records across all partitions until every task is completed.

• This is the point at which all the tasks must synchronize.

• This creates our bottleneck.

• Besides the bottleneck, this is also a significant performance hit: disk IO, network IO and more disk IO.

Once the data is shuffled, we can resume execution...

For **Stage #2**, Spark will again create a pipeline of transformations in which the shuffle data is read into RAM (Step #4c) and then perform transformations #4d, #5, #6 and finally the write action, step #7.

Lineage

From the developer's perspective, we start with a read and conclude (in this case) with a write:

Step Transformation

1 Read

2 Select

3 Filter

4 GroupBy

5 Select

6 Filter

7 Write

However, Spark starts with the action (write(..) in this case).

Next, it asks the question, what do I need to do first?

It then proceeds to determine which transformation precedes this step until it identifies the first transformation.

Step Transformation

7 Write Depends on #6

6 Filter Depends on #5

5 Select Depends on #4

4 GroupBy Depends on #3

3 Filter Depends on #2

2 Select Depends on #1

1 Read First

Why Work Backwards?

Question: So what is the benefit of working backward through your action's lineage?

Answer: It allows Spark to determine if it is necessary to execute every transformation.

Take another look at our example:

• Say we've executed this once already

• On the first execution, step #4 resulted in a shuffle

• Those shuffle files are on the various executors (src & dst)

• Because the transformations are immutable, no aspect of our lineage can change.

• That means the results of our last shuffle (if still available) can be reused.

Step Transformation

7 Write Depends on #6

6 Filter Depends on #5

5 Select Depends on #4

4 GroupBy <<< shuffle

3 Filter don't care

2 Select don't care

1 Read don't care

In this case, what we end up executing is only the operations from **Stage #2**.

This saves us the initial network read and all the transformations in **Stage #1**

Step Transformation

1 Read skipped

2 Select skipped

3 Filter skipped

4a GroupBy 1/2 skipped

4b shuffle write skipped

4c shuffle read -

4d GroupBy 2/2 -

5 Select -

6 Filter -

7 Write

And Caching...

The reuse of shuffle files (also known as our temp files) is just one example of Spark optimizing queries anywhere it can.

We cannot assume this will be available to us.

Shuffle files are by definition temporary files and will eventually be removed.

However, we cache data to explicitly accomplish the same thing that happens inadvertently with shuffle files.

In this case, the lineage plays the same role. Take for example:

Step Transformation

7 Write Depends on #6

6 Filter Depends on #5

5 Select <<< cache

4 GroupBy <<< shuffle files

3 Filter ?

2 Select ?

1 Read ?

In this case we cached the result of the select(..).

We never even get to the part of the lineage that involves the shuffle, let alone Stage #1.

Instead, we pick up with the cache and resume execution from there:

Step Transformation

1 Read skipped

2 Select skipped

3 Filter skipped

4a GroupBy 1/2 skipped

4b shuffle write skipped

4c shuffle read skipped

4d GroupBy 2/2 skipped

5a cache read -

5b Select -

6 Filter -

7 Write

https://databricks.com/blog/2015/04/28/project-tungsten-bringing-spark-closer-to-bare-metal.html

Question 25: Skipped

One of the key management features that you have at your disposal within Azure Synapse Analytics, is the ability the scale the compute resources for SQL or Spark pools to meet the demands of processing your data. Compute is separate from storage, which enables you to scale compute independently of the data in your system. This means you can scale up and scale down the compute power to meet your needs.

Apache Spark pools for Azure Synapse Analytics uses an Autoscale feature that automatically scales the number of nodes in a cluster instance up and down.

Autoscale continuously monitors the Spark instance and collects which of the following metrics? (Select five)

- ☐ Total Free Memory
(Correct)
- ☐ Total Free CPU
(Correct)
- ☐ Average Refresh rate
- ☐ Total Pending CPU
(Correct)
- ☐ Total seeds on each individual Node
- ☐ Used Memory per Node
(Correct)
- ☐ Total seeds on the Node collective
- ☐ Total number of peers on the Node network
- ☐ Total Pending Memory
(Correct)

Explanation
One of the key management features that you have at your disposal within Azure Synapse Analytics, is the ability the scale the compute resources for SQL or Spark pools to meet the demands of processing your data. In SQL pools, the unit of scale is an abstraction of compute power that is known as a data warehouse unit. Compute is separate from storage, which enables you to scale compute independently of the data in your system. This means you can scale up and scale down the compute power to meet your needs.

You can scale a Synapse SQL pool either through the Azure portal, Azure Synapse Studio or programmatically using T-SQL or PowerShell.

Scaling Apache Spark pools in Azure Synapse Analytics

Apache Spark pools for Azure Synapse Analytics uses an **Autoscale** feature that automatically scales the number of nodes in a cluster instance up and down. During the creation of a new Spark pool, a minimum and maximum number of nodes can be set

when **Autoscale** is selected. Autoscale then monitors the resource requirements of the load and scales the number of nodes up or down. To enable the Autoscale feature, complete the following steps as part of the normal pool creation process:

1. On the **Basics** tab, select the **Enable autoscale** checkbox.

2. Enter the desired values for the following properties:

• **Min** number of nodes.

• **Max** number of nodes.

The initial number of nodes will be the minimum. This value defines the initial size of the instance when it's created. The minimum number of nodes can't be fewer than three.

Autoscale continuously monitors the Spark instance and collects the following metrics:

• Total Pending CPU

The total number of cores required to start execution of all pending nodes.

• Total Pending Memory

The total memory (in MB) required to start execution of all pending nodes.

• Total Free CPU

The sum of all unused cores on the active nodes.

• Total Free Memory

The sum of unused memory (in MB) on the active nodes.

• Used Memory per Node

The load on a node. A node on which 10 GB of memory is used, is considered under more load than a worker with 2 GB of used memory.

The following conditions will then autoscale the memory or CPU

Scale-up

• Total pending CPU is greater than total free CPU for more than 1 minute.

• Total pending memory is greater than total free memory for more than 1 minute.

Scale-down

• Total pending CPU is less than total free CPU for more than 2 minutes.

• Total pending memory is less than total free memory for more than 2 minutes.

The scaling operation can take between 1 -5 minutes. During an instance where there is a scale down process, Autoscale will put the nodes in decommissioning state so that no new executors can launch on that node.

The running jobs will continue to run and finish. The pending jobs will wait to be scheduled as normal with fewer available nodes.

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/quickstart-scale-compute-portal

Question 26: Skipped

Azure Storage accounts are the base storage type within Azure. Azure Storage offers a very scalable object store for data objects and file system services in the cloud. It can also provide a messaging store for reliable messaging, or it can act as a NoSQL store.

Which of the following are Azure Storage configuration options? (Select all that apply)

- ☐ Azure Blob **(Correct)**
- ☐ Azure Cosmos DB
- ☐ Azure Queue **(Correct)**
- ☐ Azure Data Lake **(Correct)**
- ☐ Azure Table **(Correct)**
- ☐ Azure Database Server

- Azure Files
  (Correct)

Explanation
Azure Storage accounts are the base storage type within Azure. Azure Storage offers a very scalable object store for data objects and file system services in the cloud. It can also provide a messaging store for reliable messaging, or it can act as a NoSQL store.

Azure Storage offers four configuration options:

• **Azure Blob**: A scalable object store for text and binary data

• **Azure Files**: Managed file shares for cloud or on-premises deployments

• **Azure Queue**: A messaging store for reliable messaging between application components

• **Azure Table**: A NoSQL store for no-schema storage of structured data

You can use Azure Storage as the storage basis when you're provisioning a data platform technology such as Azure Data Lake Storage and HDInsight. But you can also provision Azure Storage for standalone use. For example, you provision an Azure Blob store either as standard storage in the form of magnetic disk storage or as premium storage in the form of solid-state drives (SSDs).

https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction

• **Azure Data Lake:** Microsoft Azure Data Lake is a technology in Azure cloud that enables big data analytics and artificial intelligence (AI). When this topic mentions "Data Lake," it's referring specifically to storage technology that is based on Azure Data Lake Storage Gen2.

https://docs.microsoft.com/en-us/dynamics365/fin-ops-core/dev-itpro/data-entities/azure-data-lake-overview

Question 27: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

[?] provides one-click setup, streamlined workflows, an interactive workspace for Spark-based applications plus it adds capabilities to Apache Spark, including fully managed Spark clusters and an interactive workspace.

- Azure Databricks
  (Correct)
- Azure SQL Datawarehouse
- Azure Data Factory
- Azure Data Catalogue
- Azure Cosmos DB
- Azure Storage Explorer
- Azure Data Lake Storage

Explanation
Azure Databricks

Databricks is a serverless platform that's optimized for Azure. It provides one-click setup, streamlined workflows, and an interactive workspace for Spark-based applications.

Databricks adds capabilities to Apache Spark, including fully managed Spark clusters and an interactive workspace. You can use REST APIs to program clusters.

In Databricks notebooks you'll use familiar programming tools such as R, Python, Scala, and SQL. Role-based security in Azure Active Directory and Databricks provides enterprise-grade security.

https://docs.microsoft.com/en-us/azure/databricks/scenarios/what-is-azure-databricks

Question 28: Skipped

Azure offers several types of storage for data, the one chosen should depend on the needs of the users. Each data store has a different price structure. When you want to store data but don't need to query it, which would be the most cost efficient choice?

- Azure Storage
  (Correct)
- Azure Data Lake Storage

- Azure Data Factory
- Azure Stream Analytics
- Azure Databricks
- Azure Data Catalogue

Explanation
Azure Storage offers a massively scalable object store for data objects and file system services for the cloud. There are several options for ingesting data into Azure, depending on your needs.

File storage:

• Azure Storage blobs

• Azure Data Lake Store

NoSQL databases:

• Azure Cosmos DB

• HBase on HDInsight

Analytical databases:

• Azure Data Explorer

If you create a Blob storage account, you can't directly query the data.

https://docs.microsoft.com/en-us/azure/architecture/data-guide/technology-choices/data-storage

Question 29: Skipped

During the process of creating a notebook, you need to specify the pool that needs to be attached to the notebook that is, a SQL or Spark pool.

True or False: Notebook cells are individual blocks of code or text that runs as a group. If you want to skip cells within the group, a simple skip notation in the cell is all that is required.

True

- False
(Correct)

Explanation
When you want to run notebooks in the Synapse Studio environment, you are able to run the code cells in your notebook individually or all at once. The status and progress of each cell will also be represented in the notebook.

The different functionalities for running a notebook are as follows:

• Run a Cell If you want to run once cell of code or text, you can do so through the notebook experience in Azure Synapse Studio.

• Run all cells. If you have developed code that consists of multiple cells or is combined with text, you can do so through the notebook experience in Azure Synapse Studio.

• Cancel a running cell. If you hit run, but while the cell is running, you want to cancel one cell run, you can do so within Azure Synapse notebooks.

• Cell Status indicator If you want to check the status of a cell while running or completed, you have the possibility to get a status indicator within the notebook experience in Synapse Studio.

• Spark progress indicator Azure Synapse Studio notebook is purely Spark based. Remotely, the code cells that are executed, are executed on the serverless Apache Spark pool. If you want to see the progress of a spark job, you can see in real time the job execution status below a cell. The number of tasks per each job or stage help you to identify the parallel level of your spark job. You can also drill deeper to the Spark UI of a specific job (or stage) via selecting the link on the job (or stage) name.

https://docs.microsoft.com/en-us/azure/synapse-analytics/spark/apache-spark-development-using-notebooks?tabs=classical

Question 30: Skipped

Azure Synapse Analytics is a high performing Massively Parallel Processing (MPP) engine that is built with loading and querying large datasets in mind.

There are times though when performance expectations are not met, and it is necessary then to know what aspects of the table structures and architecture can be reviewed and adapted to maximize query performance.

What is the following code intended to accomplish?

```
PowerShell
Set-AzSqlDatabase -ResourceGroupName "resourcegroupname" -DatabaseName "mySampleDataWarehouse" -ServerName "sqlpoolservername" -RequestedServiceObjectiveName "DW300c"
```

- Address the issue of low concurrency.
  (Correct)
- None of the listed options.
- Address the issue of poor query performance.
- Address the issue of poor response time.
- Address the issue of poor load performance.

Explanation
Azure Synapse Analytics is a high performing Massively Parallel Processing (MPP) engine that is built with loading and querying large datasets in mind. Many query performance enhancements are enabled by default for querying data from Azure Synapse Analytics, and additional capabilities and enhancements have both been inherited from the SQL Server product family, and have features also designed specifically to leverage the MPP capabilities within the dedicated SQL Pools architecture.

There are times though when performance expectations are not met, and it is necessary then to know what aspects of the table structures and architecture can be reviewed and adapted to maximize query performance. Symptoms that indicate that there are performance issues related to tables include:

Poor query performance

The first indication of a poor query performance issue is typically from business users who may report that their business reports are slow, or sometime not even appearing.

Poor load performance

Poor load performance may be reported by telemetry of the data loads through Azure Synapse pipelines, or you may get users reporting that the data in the reports is out of date.

Low concurrency

You may receive reports from your users that they may be unable to connect to the data warehouse to execute reports or queries.

The first response will be to ensure that the data warehouse is set to the appropriate service level range to ensure there is enough memory and concurrency slots available for multiple connections to the service. Scaling the service within the Azure portal, or Azure Synapse Studio, or issuing a Transact-SQL or the following PowerShell statement will address the issue of low concurrency.

```
PowerShell

Set-AzSqlDatabase -ResourceGroupName "resourcegroupname" -DatabaseName "mySampleDataWarehouse" -ServerName "sqlpoolservername" -RequestedServiceObjectiveName "DW300c"
```

Even with these changes, performance issue may not be resolved. Then you would have to explore other areas that we will explore in this module to resolve the issue.

https://docs.microsoft.com/en-us/azure/databricks/data/data-sources/azure/synapse-analytics

Question 31: Skipped

Azure offers a service to detect anomalies in account activities. These anomalies generate security alerts which are integrated with Azure Security Centre, and are also sent via email to subscription administrators, with details of suspicious activity and recommendations on how to investigate and remediate threats.

Which of the below is the name of this service?

- Azure Armour for Storage
- Azure Shield for Storage
- Encryption in transit
- Azure Storage Account Security Feature
- Azure Defender for Storage
  (Correct)

Explanation
Microsoft Defender for Storage detects anomalies in account activity. It then notifies you of potentially harmful attempts to access your account.

Azure Defender for Storage provides an extra layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts. This

layer of protection allows you to address threats without being a security expert or managing security monitoring systems.

Security alerts are triggered when anomalies in activity occur. These security alerts are integrated with Azure Security Centre, and are also sent via email to subscription administrators, with details of suspicious activity and recommendations on how to investigate and remediate threats.

Azure Defender for Storage is currently available for Blob storage, Azure Files, and Azure Data Lake Storage Gen2. Account types that support Azure Defender include general-purpose v2, block blob, and Blob storage accounts. Azure Defender for Storage is available in all public clouds and US government clouds, but not in other sovereign or Azure Government cloud regions.

Accounts with hierarchical namespaces enabled for Data Lake Storage support transactions using both the Azure Blob storage APIs and the Data Lake Storage APIs. Azure file shares support transactions over SMB.

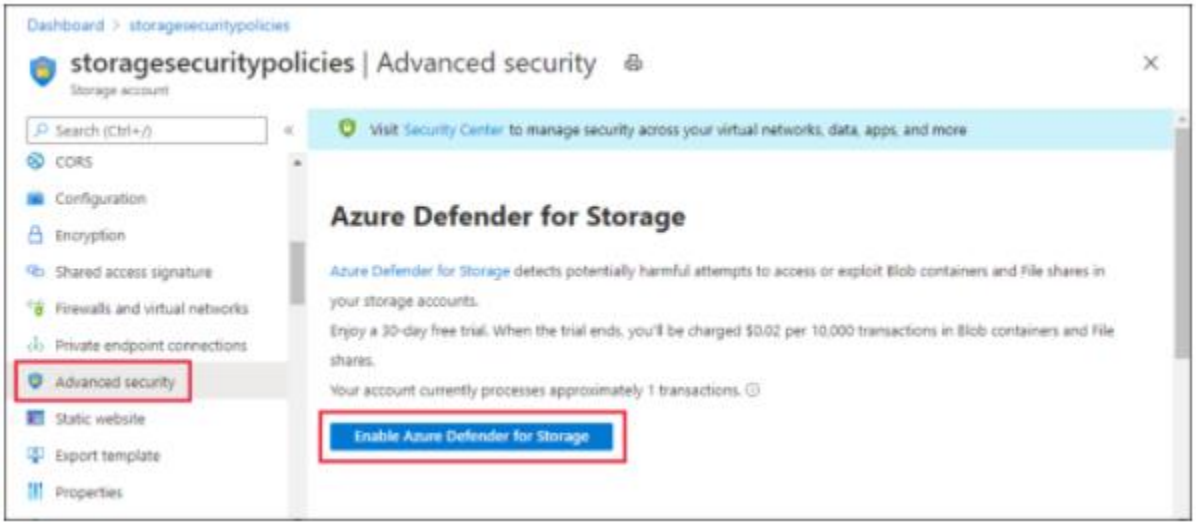
You can turn on Azure Defender for Storage in the Azure portal through the configuration page of the Azure Storage account, or in the advanced security section of the Azure portal.

Follow these steps.

1. Launch the Azure portal.

2. Navigate to your storage account. Under **Settings**, select **Advanced security**.

3. Select **Enable Azure Defender for Storage**.



https://docs.microsoft.com/en-us/azure/security-center/defender-for-storage-introduction

Question 32: Skipped

Which DataFrame method do you use to create a temporary view?

- `createOrReplaceTempView()`
  (Correct)
- `tempViewCreate()`
- `createTempViewDF()`
- `createTempView()`

Explanation
You use this method to create temporary views in DataFrames.

CREATE VIEW

Constructs a virtual table that has no physical data based on the result-set of a SQL query. `ALTER VIEW` and `DROP VIEW` only change metadata.

Syntax

```
SQL

CREATE [ OR REPLACE ] [ [ GLOBAL ] TEMPORARY ] VIEW [ IF NOT EXISTS ] view_identifier
```

`create_view_clauses AS query`

Parameters

- `OR REPLACE`

If a view of same name already exists, it is replaced.

- `[ GLOBAL ] TEMPORARY`

`TEMPORARY` views are session-scoped and is dropped when session ends because it skips persisting the definition in the underlying metastore, if any. `GLOBAL TEMPORARY` views are tied to a system preserved temporary database `global_temp`.

• `IF NOT EXISTS`

Creates a view if it does not exist.

• `view_identifier`

A view name, optionally qualified with a database name.

Syntax: `[database_name.] view_name`

• create_view_clauses

These clauses are optional and order insensitive. It can be of following formats.

• `[ ( column_name [ COMMENT column_comment ], ... ) ]` to specify column-level comments.

• `[ COMMENT view_comment ]` to specify view-level comments.

• `[ TBLPROPERTIES ( property_name = property_value [ , ... ] ) ]` to add metadata key-value pairs.

• query a `SELECT` statement that constructs the view from base tables or other views.

```
SQL
-- Create or replace view for `experienced_employee` with comments.
CREATE OR REPLACE VIEW experienced_employee
(ID COMMENT 'Unique identification number', Name)
COMMENT 'View for experienced employees'
AS SELECT id, name FROM all_employee
WHERE working_years > 5;

-- Create a global temporary view `subscribed_movies` if it does not exist.
CREATE GLOBAL TEMPORARY VIEW IF NOT EXISTS subscribed_movies
```

```
AS SELECT mo.member_id, mb.full_name, mo.movie_title

FROM movies AS mo INNER JOIN members AS mb

ON mo.member_id = mb.id;
```

https://docs.databricks.com/spark/latest/spark-sql/language-manual/sql-ref-syntax-ddl-create-view.html

Question 33: Skipped

How do you disable Azure Synapse Link for Azure Cosmos DB?

- Delete the Azure Cosmos DB container
- Delete the Azure Cosmos DB account
  (Correct)
- Set the Azure Synapse Link option to disable on the Azure Cosmos DB instance.
- Set the Azure Synapse Link option to disable on the Azure Cosmos DB container.

Explanation
After the Synapse Link capability is enabled at the account level, you cannot disable it. Understand that you will not have any billing implications if the Synapse Link capability is enabled at the account level and there is no analytical store enabled containers.

If you need to turn off the capability, you have 2 options.

• The first one is to delete and re-create a new Azure Cosmos DB account, migrating the data if necessary.

• The second option is to open a support ticket, to get help on a data migration to another account.

Deleting the Azure Cosmos DB account with disable and remove Azure Synapse Link.

https://docs.microsoft.com/en-us/azure/cosmos-db/synapse-link-frequently-asked-questions

Question 34: Skipped

Azure Synapse SQL pools support placing complex data processing logic into Stored procedures

True or False: Multiple users and client programs can perform operations on underlying database objects through a procedure, even if the users and programs do not have direct permissions on those underlying objects.

- True
  (Correct)
- False

Explanation
Azure Synapse SQL pools support placing complex data processing logic into Stored procedures. Stored procedures are great way of encapsulating one or more SQL statements or a reference to a Microsoft .NET framework Common Language Runtime (CLR) method.

Stored procedures can accept input parameters and return multiple values in the form of output parameters to the calling program. In the context of serverless SQL pools, you will perform data transformation using `CREATE EXTERNAL TABLE AS SELECT (CETAS)` statement as shown in the following example.

```
SQL

-- this sample references external data source and external file format defined i
n previous section

CREATE PROCEDURE usp_calculate_population_by_year_state

AS

BEGIN

CREATE EXTERNAL TABLE population_by_year_state

WITH (

LOCATION = 'population_by_year_state/',

DATA_SOURCE = destination_ds,

FILE_FORMAT = parquet_file_format

)

AS

SELECT decennialTime, stateName, SUM(population) AS population

FROM

OPENROWSET(BULK 'https://azureopendatastorage.blob.core.windows.net/censusdatacon
tainer/release/us_population_county/year=*/*.parquet',

FORMAT='PARQUET') AS [r]

GROUP BY decennialTime, stateName

END
```

```
GO
```

In addition to encapsulating Transact-SQL logic, stored procedures also provide the following additional benefits:

Reduces client to server network traffic

The commands in a procedure are executed as a single batch of code. This can significantly reduce network traffic between the server and client because only the call to execute the procedure is sent across the network.

Provides a security boundary

Multiple users and client programs can perform operations on underlying database objects through a procedure, even if the users and programs do not have direct permissions on those underlying objects. The procedure controls what processes and activities are performed and protects the underlying database objects. This eliminates the requirement to grant permissions at the individual object level and simplifies the security layers.

Eases maintenance

When client applications call procedures and keep database operations in the data tier, only the procedures must be updated for any changes in the underlying database.

Improved performance

Stored procedures are compiled the first time they are executed, and the subsequent execution plan is held in the cache and reused on subsequent execution of the same stored procedure. As a result, it takes less time to process the procedure.

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql/develop-stored-procedures

Question 35: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

Azure Synapse Analytics can work by acting as the one stop shop to meet all of your analytical needs in an integrated environment if you do not have an analytical environment in place already.

[?] is a single web UI that allows you to:

• Explore your data estate.

• Develop TSQL scripts and notebooks to interact with the analytical engines.

• Build data integration pipelines for managing data movement.

• Monitor the workloads within the service.

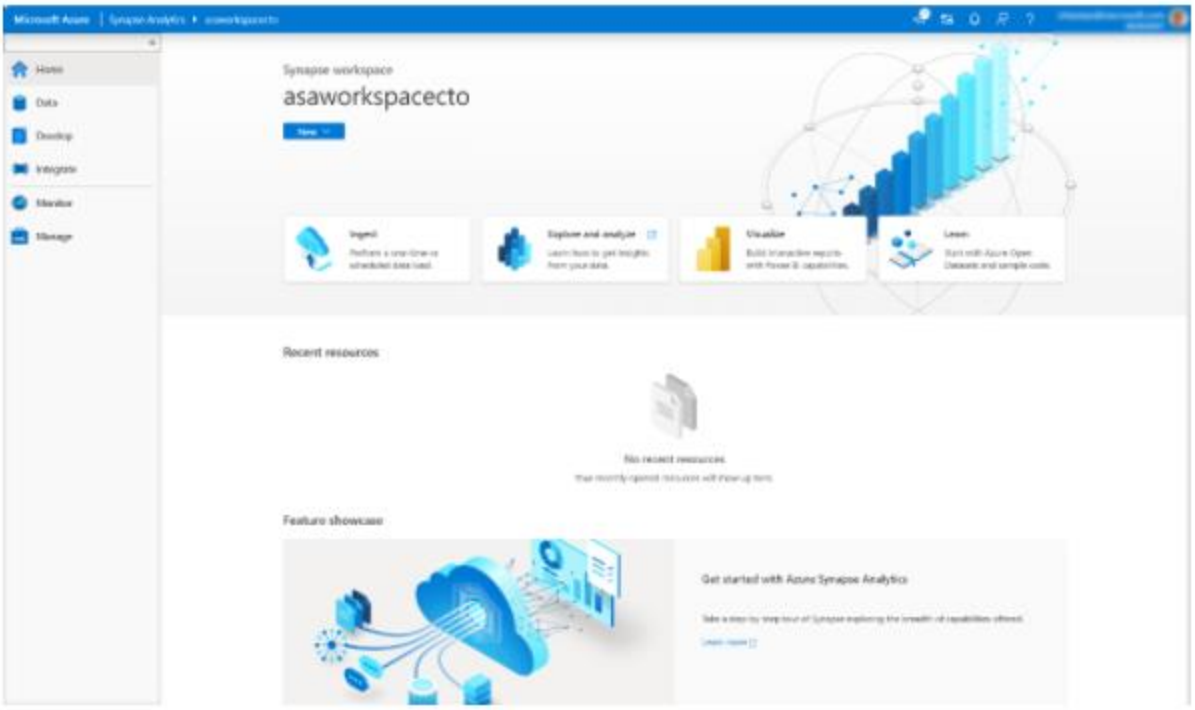
• Manage the components of the service.

- Azure Synapse Studio
(Correct)
- Azure Portal
- Azure Monitor
- Azure Designer
- Azure Pipelines
- Azure DevOps

Explanation
Azure Synapse Analytics can work by acting as the one stop shop to meet all of your analytical needs in an integrated environment if you do not have an analytical environment in place already.

A single Web UI to be able to access all Azure Synapse Analytics capabilities

While the Azure Portal will allow you to manage some aspects of the product, Azure Synapse Studio is the best place to centrally work with all the capabilities.



Azure Synapse Studio is a single web UI that allows you to:

• Explore your data estate.

• Develop TSQL scripts and notebooks to interact with the analytical engines.

• Build data integration pipelines for managing data movement.

• Monitor the workloads within the service.

• Manage the components of the service.

https://docs.microsoft.com/en-us/azure/synapse-analytics/overview-what-is

Question 36: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

As a Data Engineer, you can transfer and move data in several ways. The most common tool is Azure Data Factory which provides robust resources and nearly 100 enterprise connectors. Azure Data Factory also allows you to transform data by using a wide variety of languages.

Azure has opened the way for technologies that can handle unstructured data at an unlimited scale. This change has shifted the paradigm for loading and transforming data from [?].

- ETL → MTD
- ETL → RTO
- ETL → ELT
  (Correct)
- RPO → RTO
- ELT → ETL
- MTD → RPO

Explanation
As a Data Engineer, you can transfer and move data in several ways. One way is to start an *Extract, Transform, and Load (ETL)* process.

Extraction sources can include databases, files, and streams. Each source has unique data formats that can be structured, semistructured, or unstructured. In Azure, data sources include Azure Cosmos DB, Azure Data Lake, files, and Azure Blob storage.

ETL tools

As a data engineer, you'll use several tools for ETL. The most common tool is Azure Data Factory, which provides robust resources and nearly 100 enterprise connectors. Data Factory also allows you to transform data by using a wide variety of languages.

You might find that you also need a repository to maintain information about your organization's data sources and dictionaries. Azure Data Catalogue can store this information centrally.

Azure Data Factory

Data Factory is a cloud-integration service. It orchestrates the movement of data between various data stores.

As a data engineer, you can create data-driven workflows in the cloud to orchestrate and automate data movement and data transformation. Use Data Factory to create and schedule data-driven workflows (called pipelines) that can ingest data from data stores.

Data Factory processes and transforms data by using compute services such as Azure HDInsight, Hadoop, Spark, and Azure Machine Learning. Publish output data to data stores such as Azure SQL Data Warehouse so that business intelligence applications can consume the data. Ultimately, you use Data Factory to organize raw data into meaningful data stores and data lakes so your organization can make better business decisions.

https://docs.microsoft.com/en-us/azure/data-factory/introduction

Evolution from ETL

Azure has opened the way for technologies that can handle unstructured data at an unlimited scale. This change has shifted the paradigm for loading and transforming data from ETL to extract, load, and transform (ELT).

The benefit of ELT is that you can store data in its original format, be it JSON, XML, PDF, or images. In ELT, you define the data's structure during the transformation phase, so you can use the source data in multiple downstream systems.

In an ELT process, data is extracted and loaded in its native format. This change reduces the time required to load the data into a destination system. The change also limits resource contention on the data sources.

The steps for the ELT process are the same as for the ETL process. They just follow a different order.

Another process like ELT is called extract, load, transform, and load (ELTL). The difference with ELTL is that it has a final load into a destination system.

https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl

Question 37: Skipped

What type of process are the driver and the executors?

- JavaScript
- C++ processes
- Python processes
- Java processes
  (Correct)

Explanation
The driver and the executors are Java processes.

What is a JVM?

The JVM manages system memory and provides a portable execution environment for Java-based applications

Technical definition: The JVM is the specification for a software program that executes code and provides the runtime environment for that code.

Everyday definition: The JVM is how we run our Java programs. We configure the JVM's settings and then rely on it to manage program resources during execution.

The **Java Virtual Machine (JVM)** is a program whose purpose is to execute other programs.

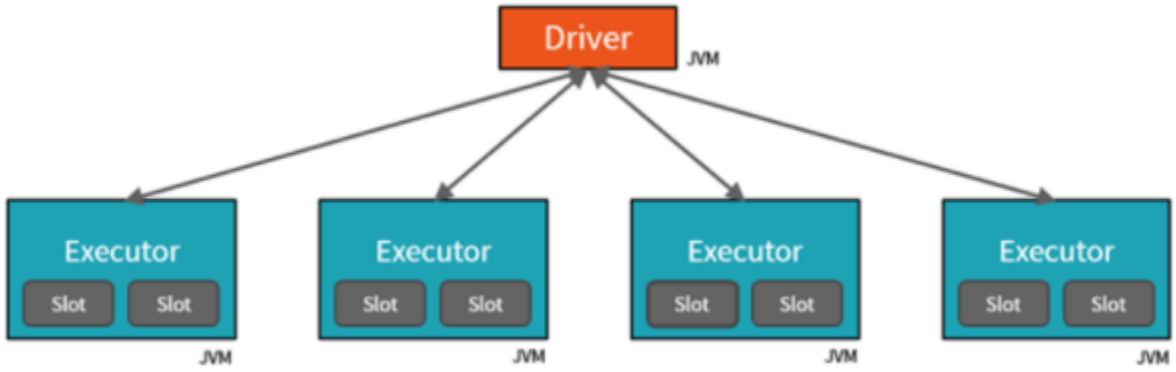
The JVM has **two primary functions**:

1.To allow Java programs to run on any device or operating system (known as the *"Write once, run anywhere"* principle)

2.To manage and optimize program memory

JVM view of the Spark Cluster: *Drivers, Executors, Slots & Tasks*

The Spark runtime architecture leverages JVMs:



https://www.rakirahman.me/spark-certification-study-guide-part-1/

Question 38: Skipped

Scenario: Queen Consolidated was overtaken by Raymond Carson Palmer and re-branded as Palmer Technologies. Now that Ray is overseeing the operations at Palmer, Ray has decided to implement better applications.

You are working as a consultant with Palmer, and in a meeting with Ray and his IT team discussing Azure Data Factory. The team plans to use Azure Data Factory to prepare data to be queried by Azure Synapse Analytics serverless SQL pools.

Files will be initially ingested into an Azure Data Lake Storage Gen2 account as 10 small JSON files. Each file will contain the same data attributes and data from a subsidiary of Palmer. The team needs to move the files to a different folder and transform the data.

Required:
• Provide the fastest possible query times.
• Automatically infer the schema from the underlying files.

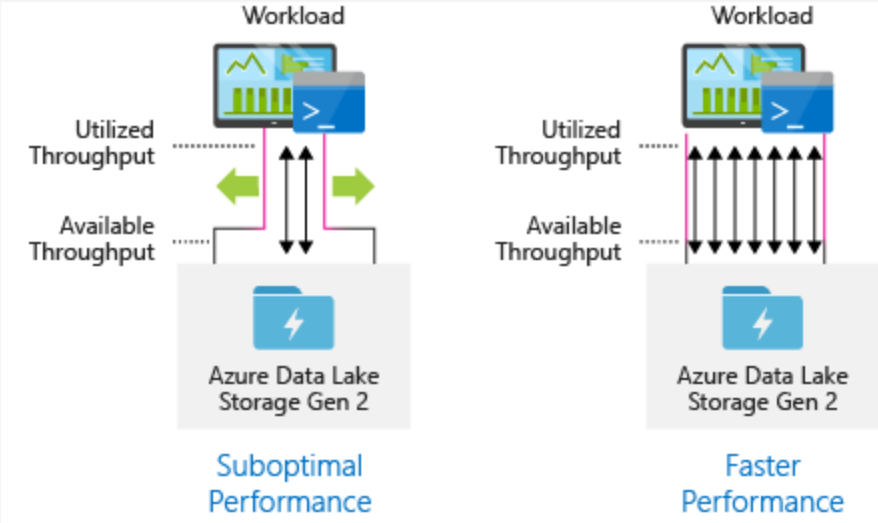
As the Azure expert, the team looks to you for advice on how they should configure the Data Factory copy activity with respect to the copy behaviour.

Which of the following should you advise them to use?

- Preserve hierarchy
- Append Files
- Flatten hierarchy
- Merge Files
  (Correct)

Explanation
With respect to the copy behaviour, you should advise the team to Merge files.

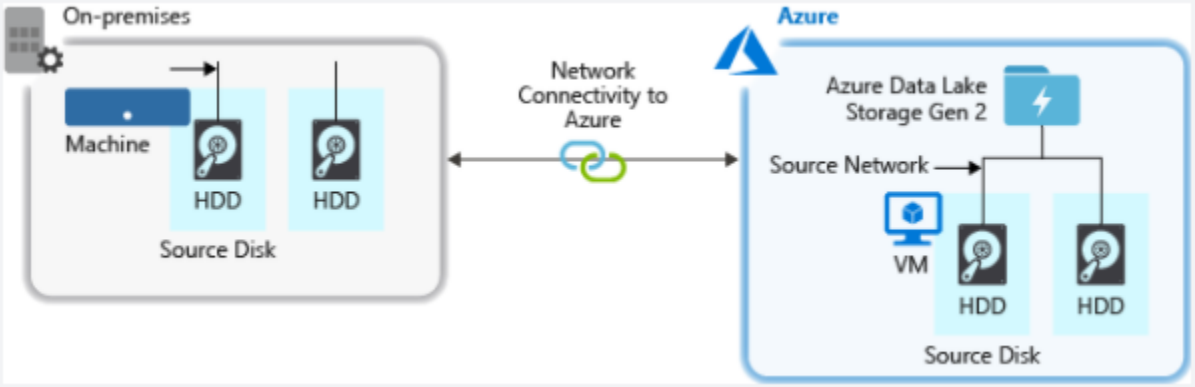
Azure Data Lake Storage Gen2 supports high-throughput for I/O intensive analytics and data movement. In Data Lake Storage Gen2, using all available throughput – the amount of data that can be read or written per second – is important to get the best performance. This is achieved by performing as many reads and writes in parallel as possible.



Data Lake Storage Gen2 can scale to provide the necessary throughput for all analytics scenarios. By default, a Data Lake Storage Gen2 account provides enough throughput in its default configuration to meet the needs of a broad category of use cases. For the cases where customers run into the default limit, the Data Lake Storage Gen2 account can be configured to provide more throughput by contacting Azure Support.

Data ingestion

When ingesting data from a source system to Data Lake Storage Gen2, it is important to consider that the source hardware, source network hardware, or network connectivity to Data Lake Storage Gen2 can be the bottleneck.



It is important to ensure that the data movement is not affected by these factors.

Optimizing I/O intensive jobs on Hadoop and Spark workloads on HDInsight

Jobs fall into one of the following three categories:

CPU intensive. These jobs have long computation times with minimal I/O times. Examples include machine learning and natural language processing jobs.

Memory intensive. These jobs use lots of memory. Examples include PageRank and real-time analytics jobs.

I/O intensive. These jobs spend most of their time doing I/O. A common example is a copy job which does only read and write operations. Other examples include data preparation jobs that read a lot of data, performs some data transformation, and then writes the data back to the store.

The following guidance is only applicable to I/O intensive jobs.

General considerations

You can have a job that reads or writes as much as 100MB in a single operation, but a buffer of that size might compromise performance. To optimize performance, try to keep the size of an I/O operation between 4MB and 16MB.

General considerations for an HDInsight cluster

HDInsight versions. For best performance, use the latest release of HDInsight.

Regions. Place the Data Lake Storage Gen2 account in the same region as the HDInsight cluster.

An HDInsight cluster is composed of two head nodes and some worker nodes. Each worker node provides a specific number of cores and memory, which is determined by the VM-type. When running a job, YARN is the resource negotiator that allocates the available memory and cores to create containers. Each container runs the tasks needed to complete the job. Containers run in parallel to process tasks quickly. Therefore, performance is improved by running as many parallel containers as possible.

There are three layers within an HDInsight cluster that can be tuned to increase the number of containers and use all available throughput.

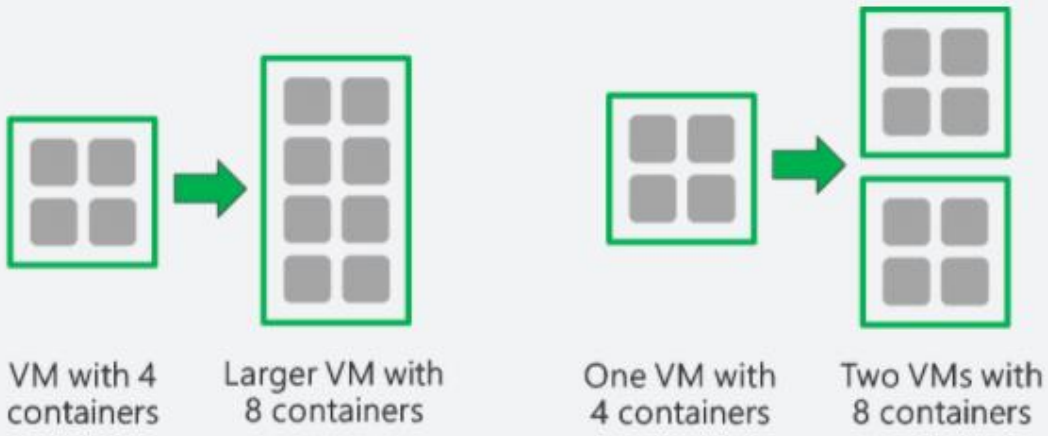
Physical layer

YARN layer

Workload layer

Physical Layer

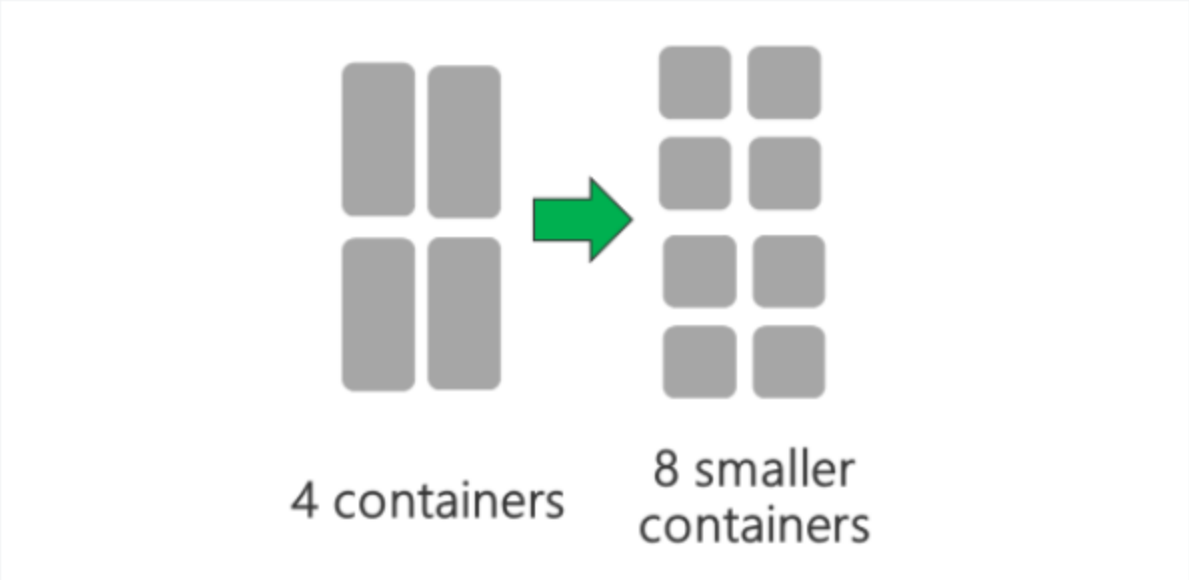
Run cluster with more nodes and/or larger sized VMs. A larger cluster will enable you to run more YARN containers as shown in the picture below.



Use VMs with more network bandwidth. The amount of network bandwidth can be a bottleneck if there is less network bandwidth than Data Lake Storage Gen2 throughput. Different VMs will have varying network bandwidth sizes. Choose a VM-type that has the largest possible network bandwidth.

YARN Layer

Use smaller YARN containers. Reduce the size of each YARN container to create more containers with the same amount of resources.

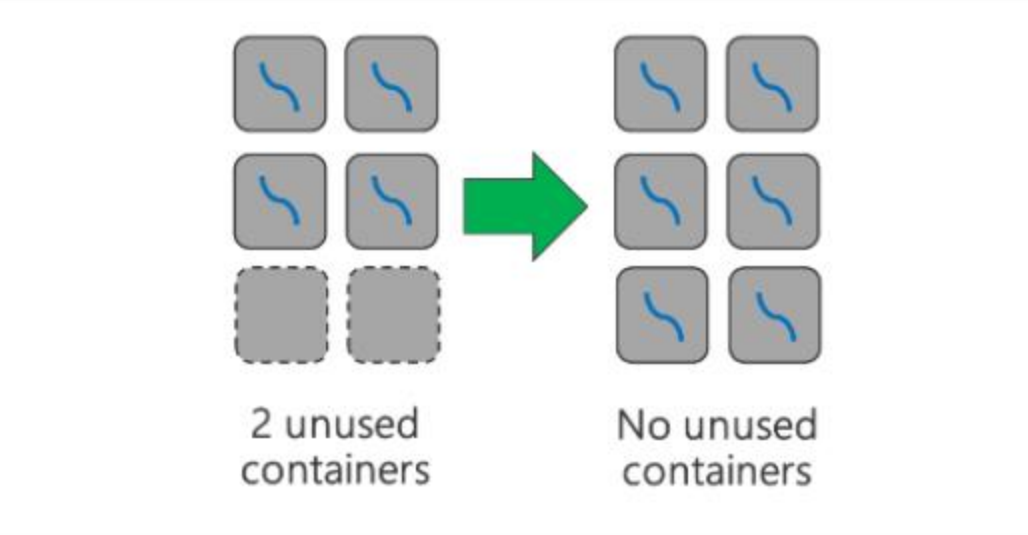


Depending on your workload, there will always be a minimum YARN container size that is needed. If you pick too small a container, your jobs will run into out-of-memory issues. Typically YARN containers should be no smaller than 1GB. It's common to see 3GB YARN containers. For some workloads, you may need larger YARN containers.

Increase cores per YARN container. Increase the number of cores allocated to each container to increase the number of parallel tasks that run in each container. This works for applications like Spark which run multiple tasks per container. For applications like Hive which run a single thread in each container, it is better to have more containers rather than more cores per container.

Workload Layer

Use all available containers. Set the number of tasks to be equal or larger than the number of available containers so that all resources are utilized.



Failed tasks are costly. If each task has a large amount of data to process, then failure of a task results in an expensive retry. Therefore, it is better to create more tasks, each of which processes a small amount of data.

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-performance-tuning-guidance

Question 39: Skipped

SQL Server Integration Services (SSIS) is a platform for building complex Extract Transform and Load (ETL) solutions. SSIS is a component within SQL Server and consists of a Windows service that manages the execution of ETL workflows, along with several tools and components for developing those workflows.

SSIS is primarily a control flow engine that manages the execution of workflows. Workflows are held in packages, which can be executed [?]. (Select all that apply)

- ☐ On a schedule **(Correct)**
- ☐ Only once **(Correct)**
- ☐ Randomly
- ☐ On demand **(Correct)**

Explanation
SQL Server Integration Services (SSIS) is a platform for building complex Extract Transform and Load (ETL) solutions. SSIS is a component within SQL Server and consists of a Windows service that manages the execution of ETL workflows, along with several tools and components for developing those workflows. It is typically used to develop data integration pipelines for on-premises data warehousing solutions. It can also be used to create data migration pipelines when migrating data between different systems.

SSIS is primarily a control flow engine that manages the execution of workflows. Workflows are held in packages, which can be executed on demand, or on a schedule (including only a single run by using the trigger now feature). Development of SSIS packages, the task workflow is referred to as the control flow of the package. A control flow can include a specific task to manage data flow operations. SSIS executes these Data Flow tasks by using a data flow engine that encapsulates the data flow in a pipeline. Each step in the Data Flow task operates in sequence on a rowset of data as it passes through the pipeline.

A SSIS solution usually consists of one or more SSIS projects, each containing one or more SSIS packages.

SSIS projects

From SQL Server 2012, a project is the unit of deployment for SSIS solutions. You can define project-level parameters to enable users to specify run-time settings, and project-level connection managers that reference data sources and destinations used in package data flows. You can then deploy projects to an SSIS Catalogue in a SQL Server instance, and configure project-level parameter values and connections as appropriate for execution environments.

SSIS packages

A project contains one or more packages, each defining a workflow of tasks to be executed. The workflow of tasks in a package is referred to as its control flow. A package control flow can include one or more Data Flow task, each of which encapsulates its own data flow pipeline. Packages can include package-level parameters so that dynamic values can be passed to the package at run time. In previous releases of SSIS, deployment was managed at the package level.

https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services?view=sql-server-ver15

Question 40: Skipped

To provide a better authoring experience, Azure Data Factory allows you to configure version control software for easier change tracking and collaboration. Which of the below does Azure Data Factory integrate with? (Select all that apply)

- [ ] Google Cloud Source Repositories
- [ ] Team Foundation Server
- [ ] AWS CodeCommit
- [ ] Launchpad
- [ ] Source Safe
- [x] Git repositories
(Correct)
- [ ] GitLab
- [ ] BitBucket
- [ ] SourceForge

Explanation
Azure Data Factory allows you to configure a Git repository with either Azure Repos or GitHub, and is a version control system that allows for easier change tracking and collaboration.

https://docs.microsoft.com/en-us/azure/data-factory/source-control

Question 41: Skipped

Scenario: The organization you work at has data which is specific to a country or region due to regulatory control requirements.

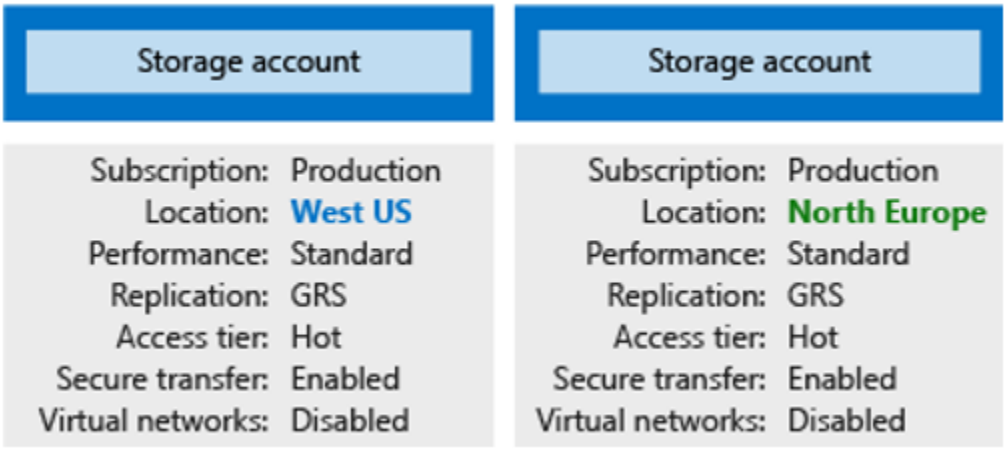
When considering Azure Storage Accounts, which option meets the data diversity requirement?

- Locate the organization's data it in a data centre with the strictest data regulations to ensure that regulatory requirement thresholds have been met. In this way, only one storage account will be required for managing all data, which will reduce data storage costs.

- Locate the organization’s data it in a data centre in the required country or region with one storage account for each location.
  (Correct)
- Enable virtual networks for the proprietary data and not for the public data. This will require separate storage accounts for the proprietary and public data.
- None of the listed options.

Explanation
How many storage accounts do you need?

A storage account represents a collection of settings like location, replication strategy, and subscription owner. You need one storage account for every group of settings that you want to apply to your data. The following illustration shows two storage accounts that differ in one setting; that one difference is enough to require separate storage accounts.



The number of storage accounts you need is typically determined by your data diversity, cost sensitivity, and tolerance for management overhead.

Data diversity

Organizations often generate data that differs in where it is consumed, how sensitive it is, which group pays the bills, etc. Diversity along any of these vectors can lead to multiple storage accounts. Let's consider two examples:

1. Do you have data that is specific to a country or region? If so, you might want to locate it in a data centre in that country for performance or compliance reasons. You will need one storage account for each location.

2. Do you have some data that is proprietary and some for public consumption? If so, you could enable virtual networks for the proprietary data and not for the public data. This will also require separate storage accounts.

In general, increased diversity means an increased number of storage accounts.

https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

Question 42: Skipped

Scenario: A teammate is working on solution for transferring data between a dedicated SQL Pool and a serverless Apache Spark Pool using the Azure Synapse Apache Spark Pool to Synapse SQL connector.

When could SQL Auth be used for this connection?

- When you need a token-based authentication to a dedicated SQL outside of the Synapse Analytics workspace.
  (Correct)
- None of the listed options.
- Never, it is not necessary to use SQL Auth when transferring data between a SQL or Spark Pool.
- Always, anytime you want to transfer data between the SQL and Spark Pool.

Explanation
Currently, the Azure Synapse Apache Spark Pool to Synapse SQL connector does not support a token-based authentication to a dedicated SQL pool that is outside of the workspace of Synapse Analytics. In order to establish and transfer data to a dedicated SQL pool that is outside of the workspace without Azure AD, you would have to use SQL Auth.

https://social.technet.microsoft.com/wiki/contents/articles/53259.azure-sql-three-ways-to-copy-databases-between-azure-sql-servers.aspx

Question 43: Skipped

True or False: In Azure Data Factory, in order to debug pipelines or activities, it is necessary to publish your workflows. Pipelines or activities which are being tested may be confined to containers to isolate them from the production environment.

- False
  (Correct)
- True

Explanation

Customer requirements and expectations are changing in relation to data integration. The need among users to develop and debug their Extract Transform/Load (ETL) and Extract Load/Transform (ELT) workflows iteratively is therefore becoming more imperative.

Azure Data Factory can help you build and develop iterative debug Data Factory pipelines when you develop your data integration solution. By authoring a pipeline using the pipeline canvas, you can test your activities and pipelines by using the Debug capability.

In Azure Data Factory, there is no need to publish changes in the pipeline or activities before you want to debug. This is helpful in a scenario where you want to test the changes and see if it works as expected before you actually save and publish them.

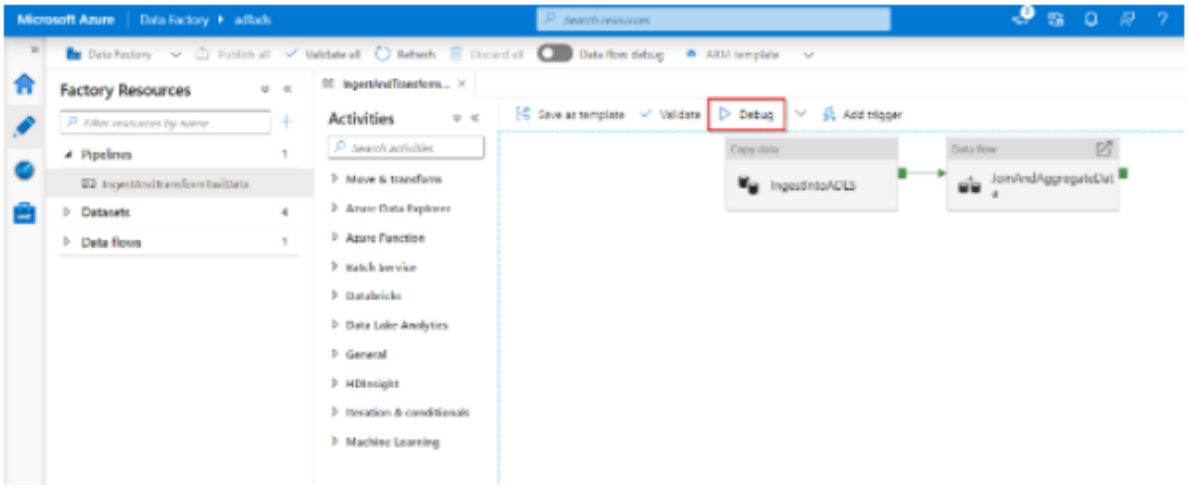
Sometimes, you don't want to debug the whole pipeline but test a part of the pipeline. A Debug run allows you to do just that. You can test the pipeline end to end or set a breakpoint. By doing so in debug mode, you can interactively see the results of each step while you build and debug your pipeline.

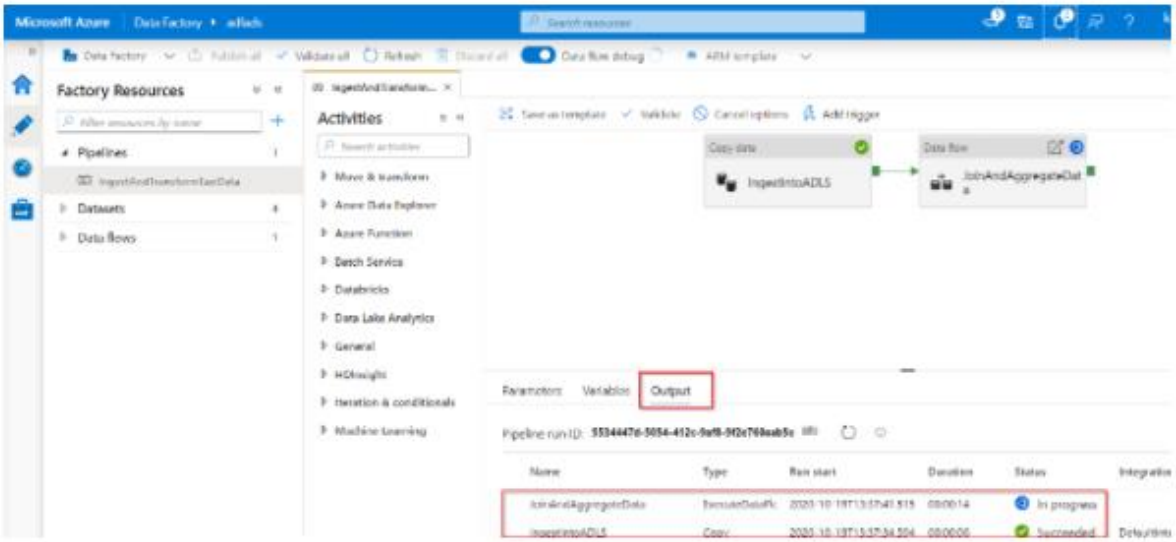
Debug and publish a pipeline:

As you create or modify a pipeline that is running, you can see the results of each activity in the Output tab of the pipeline canvas.

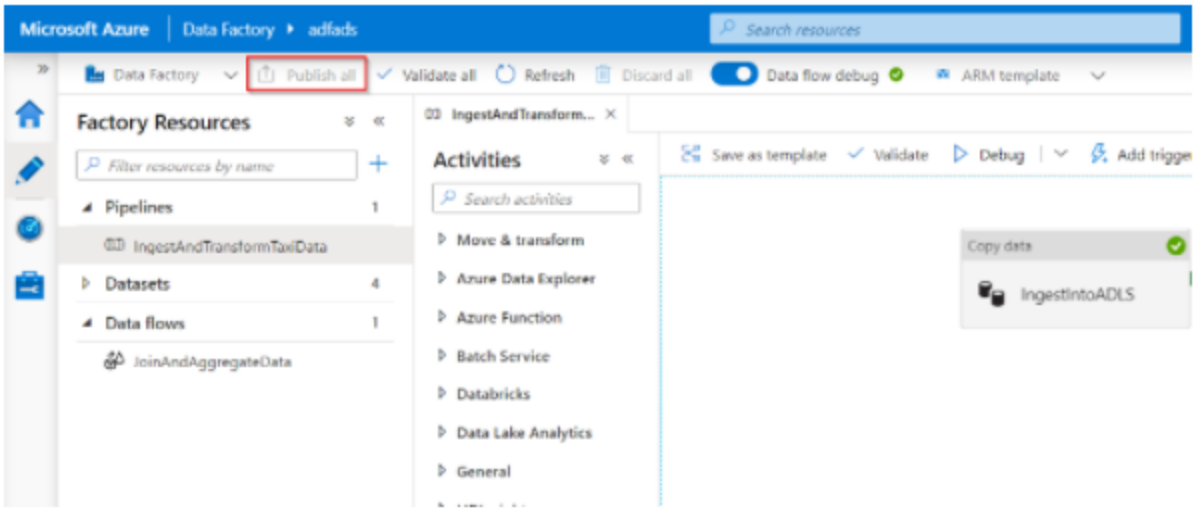
After a test run succeeds, and you are satisfied with the results, you can add more activities to the pipeline and continue debugging in an iterative manner. When you are not satisfied or like to stop the pipeline from debugging, you can cancel a test run while it is in progress. You do need to be aware that by selecting the debug slider, it will actually run the pipeline. Therefore, if the pipeline contains, for example, a copy activity, the test run will copy data from source to destination. A best practice is to use test folders in your copy activities and other activities when debugging such that when you are satisfied with the results and have debugged the pipeline, you switch to the actual folders for your normal operations.

1. To debug the pipeline, select Debug on the toolbar. You see the status of the pipeline run in the Output tab at the bottom of the window.

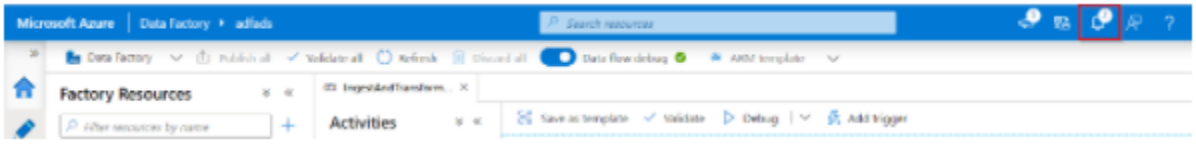




2. Once the pipeline can run successfully, in the top toolbar, select Publish all. This action publishes entities (datasets, and pipelines) you created to Data Factory.



3. Wait until you seetheSuccessfully published message. To see notification messages, click theShow Notifications on the top-right (bell button).



https://docs.microsoft.com/en-us/azure/data-factory/iterative-development-debugging

Question 44: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

[?] is a data flow object that can be added to the canvas designer as an activity in an Azure Data Factory pipeline to perform code free data preparation. It enables individuals who are not conversant with the traditional data preparation technologies such as Spark or SQL Server, and languages such as Python and T-SQL to prepare data at cloud scale iteratively.

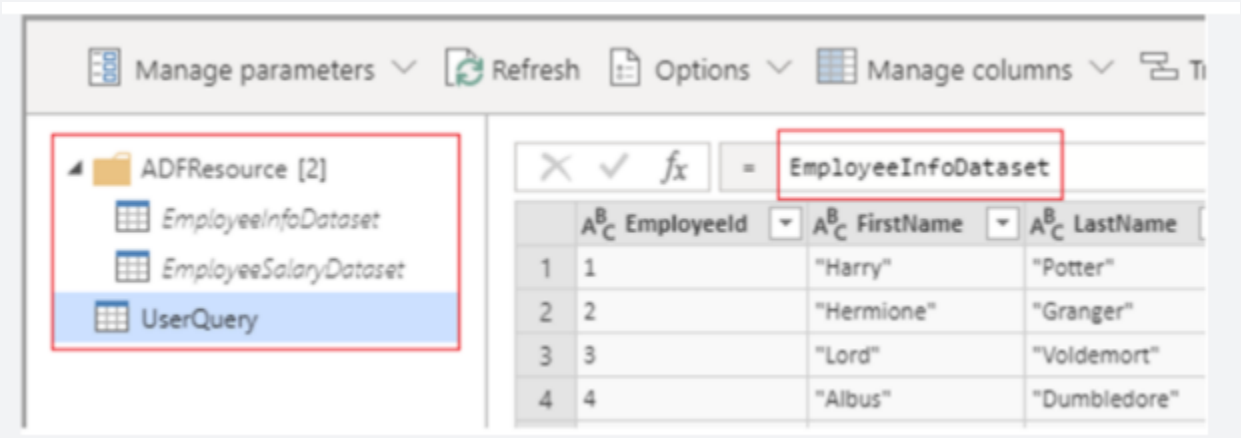
- Wrangling Data Flow
  (Correct)

- Data Stream Expression Builder
- Mapping Data Flow
- Data Expression Script Builder
- Data Expression Orchestrator
- Data Flow Expression Builder

Explanation

Wrangling Data Flow is a data flow object that can be added to the canvas designer as an activity in an Azure Data Factory pipeline to perform code free data preparation. It enables individuals who are not conversant with the traditional data preparation technologies such as Spark or SQL Server, and languages such as Python and T-SQL to prepare data at cloud scale iteratively.

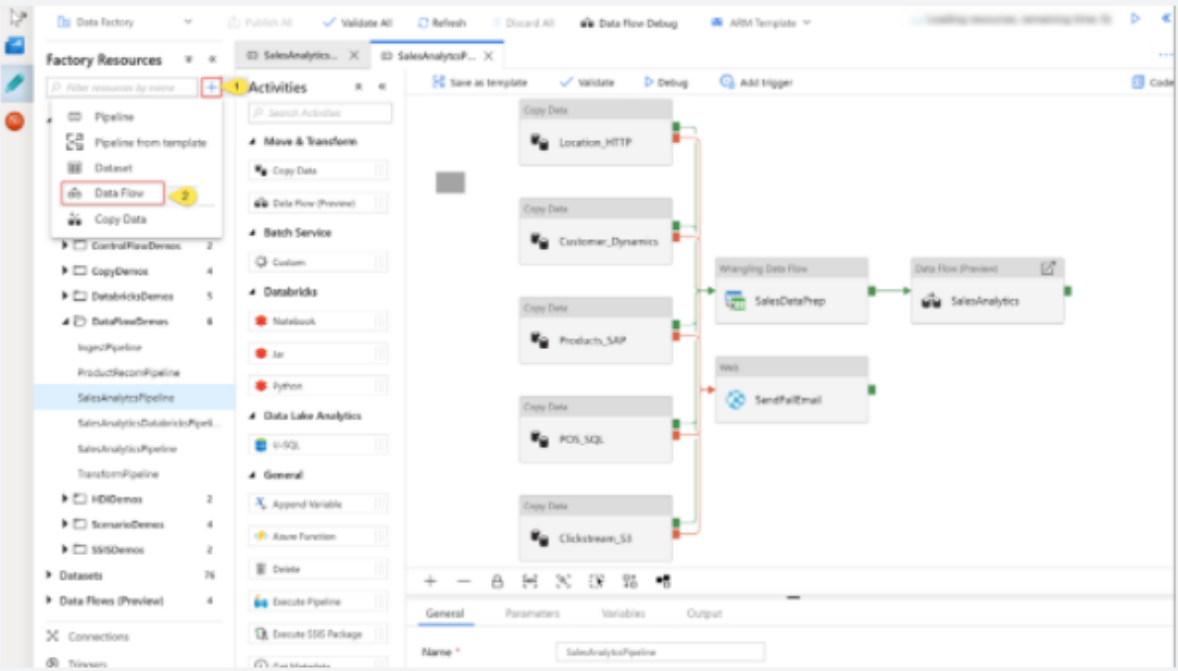
The Wrangling Data Flow uses a grid type interface for basic data preparation that is like the aesthetics of Excel, known as an Online Mashup Editor. The editor also enables more advanced users to perform more complex data preparation using formulas.



The formulas work with Power Query Online and makes Power Query M functions available for data factory users. Wrangling data flow then translates the M language generated by the Power Query Online Mashup Editor into spark code for cloud scale execution.

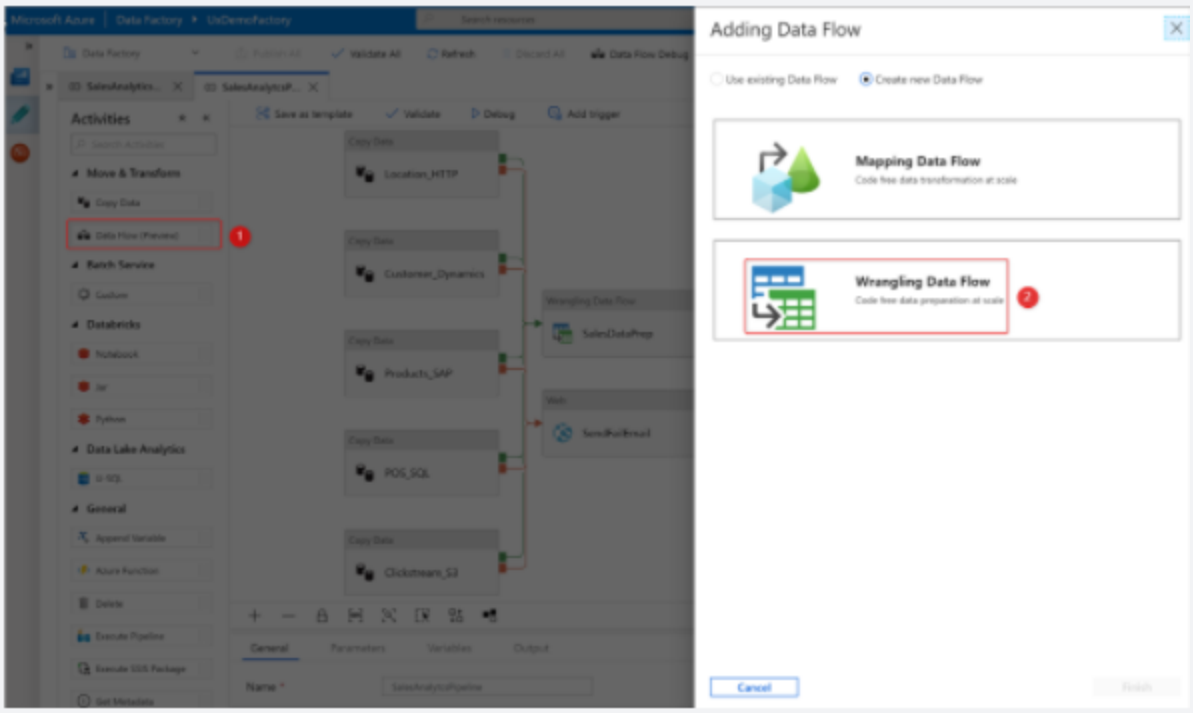
This capability enables both data engineers and citizen data integrators to interactively explore and prepare datasets. In addition, they can interactively work with the M language and preview the result before viewing it in the context of a wider pipeline.

There are two ways to create a wrangling data flow in Azure Data Factory. One way is to click the plus icon and select Data Flow in the factory resources pane.



The other method is in the activities pane of the pipeline canvas. Open the Move and Transform accordion and drag the Data flow activity onto the canvas.

In both methods, in the side pane that opens, select Create new data flow and choose Wrangling data flow. Click OK.



Add a Source dataset for your wrangling data flow, and select a sink dataset. The following data sources are supported.

Connector: Azure Blob Storage

Data format & Authentication type: CSV, Parquet | Account Key

Connector: Azure Data Lake Storage Gen1

Data format & Authentication type: CSV | Service Principal

Connector: Azure Data Lake Storage Gen2

Data format & Authentication type: CSV, Parquet | Account Key, Service Principal

Connector: Azure SQL Database

Data format & Authentication type: N/A | SQL authentication

Connector: Azure Synapse Analytics

Data format & Authentication type: N/A | SQL authentication

https://docs.microsoft.com/en-us/azure/data-factory/wrangling-overview

Question 45: Skipped

Scenario: You are working as a consultant at Advanced Idea Mechanics (A.I.M.) who is a privately funded think tank organized of a group of brilliant scientists whose sole dedication is to acquire and develop power through technological means. Their goal is to use this power to overthrow the governments of the world. They supply arms and technology to radicals and subversive organizations in order to foster a violent technological revolution of society while making a profit.

The company has 10,000 employees. Most employees are located in Europe. The company supports teams worldwide.

AIM has two main locations: a main office in London, England, and a manufacturing plant in Berlin, Germany.

At the moment, you are leading a Workgroup meeting with the IT Team where the topic of discussion is Azure Databricks.

The IT team plans to create an Azure Databricks workspace that has a tiered structure. The workspace will contain the following three workloads:
• A workload for data engineers who will use Python and SQL.
• A workload for jobs that will run notebooks that use Python, Scala, and SOL.
• A workload that data scientists will use to perform ad hoc analysis in Scala and R.

The enterprise architecture team at AIM identifies the following standards for Databricks environments:
• The data engineers must share a cluster.
• The job cluster will be managed by using a request process whereby data scientists and data engineers provide packaged notebooks for deployment to the cluster.
• All the data scientists must be assigned their own cluster that terminates automatically after 120 minutes of inactivity. Currently, there are three data scientists.

Required: Create the Databricks clusters for the workloads.

Solution: The team decides to create a High Concurrency cluster for each data scientist, a High Concurrency cluster for the data engineers, and a Standard cluster for the jobs.

Does this meet the requirement?

- No
(Correct)
- Yes

Explanation
High-concurrency clusters do not support Scala.

Standard clusters

Standard clusters are recommended for a single user. Standard clusters can run workloads developed in any language: Python, R, Scala, and SQL.

High Concurrency clusters

A High Concurrency cluster is a managed cloud resource. The key benefits of High Concurrency clusters are that they provide Apache Spark-native fine-grained sharing for maximum resource utilization and minimum query latencies.

High Concurrency clusters work only for SQL, Python, and R. The performance and security of High Concurrency clusters is provided by running user code in separate processes, which is not possible in Scala.

In addition, only High Concurrency clusters support table access control.

https://docs.microsoft.com/en-us/azure/databricks/clusters/configure

Question 46: Skipped

Scenario: Dr. Karl Malus works for the Power Broker Corporation (PBC) founded by Curtiss Jackson, using technology to service various countries and their military efforts. You have been contracted by the company to assist Dr. Malus with their Microsoft Azure Databricks projects.

The team plans to create an Azure Databricks workspace that has a tiered structure. The workspace will contain the following three workloads:
• A workload for data engineers who will use Python and SQL.
• A workload for jobs that will run notebooks that use Python, Scala, and SOL.
• A workload that data scientists will use to perform ad hoc analysis in Scala and R.

The enterprise architecture team at PBC identifies the following standards for Databricks environments:
• The data engineers must share a cluster.
• The job cluster will be managed by using a request process whereby data scientists and data engineers provide packaged notebooks for deployment to the cluster.
• All the data scientists must be assigned their own cluster that terminates automatically after 120 minutes of inactivity. Currently, there are three data scientists.

Required: The team needs to create the Databricks clusters for the workloads.

Solution: The team creates a Standard cluster for each data scientist, a High Concurrency cluster for the data engineers, and a High Concurrency cluster for the jobs.

Does this meet the requirement?

- No
  (Correct)
- Yes

Explanation
The solution does not meet the requirement because: "High Concurrency clusters work only for SQL, Python, and R. The performance and security of High Concurrency clusters is provided by running user code in separate processes, which is not possible in Scala."

Standard clusters

Standard clusters are recommended for a single user. Standard clusters can run workloads developed in any language: Python, R, Scala, and SQL.

High Concurrency clusters

A High Concurrency cluster is a managed cloud resource. The key benefits of High Concurrency clusters are that they provide Apache Spark-native fine-grained sharing for maximum resource utilization and minimum query latencies.

High Concurrency clusters work only for SQL, Python, and R. The performance and security of High Concurrency clusters is provided by running user code in separate processes, which is not possible in Scala.

In addition, only High Concurrency clusters support table access control.

https://docs.microsoft.com/en-us/azure/databricks/clusters/configure

Question 47: Skipped

Azure Storage provides a REST API to work with the containers and data stored in each account.

See the below command:

```
HTTP
GET https://[url-for-service-account]/?comp=list&include=metadata
```

What would this command return?

- All the listed options.
- A list all the blobs in a container **(Correct)**
- A list all the tables in a container
- A list all the files in a container
- A list all the queues in a container
- None of the listed options.

Explanation
Azure Storage provides a REST API to work with the containers and data stored in each account. There are independent APIs available to work with each type of data you can store. We have four specific data types:

• **Blobs** for unstructured data such as binary and text files.

• **Queues** for persistent messaging.

• **Tables** for structured storage of key/values.

• **Files** for traditional SMB file shares.

Use the REST API

The Storage REST APIs are accessible from anywhere on the Internet, by any app that can send an HTTP/HTTPS request and receive an HTTP/HTTPS response.

If you wanted to list all the blobs in a container, you would send something like:

```
HTTP
GET https://[url-for-service-account]/?comp=list&include=metadata
```

This would return an XML block with data specific to the account:

```
XML
<?xml version="1.0" encoding="utf-8"?>
<EnumerationResults AccountName="https://[url-for-service-account]/">
<Containers>
<Container>
<Name>container1</Name>
<Url>https://[url-for-service-account]/container1</Url>
<Properties>
<Last-Modified>Sun, 24 Sep 2018 18:09:03 GMT</Last-Modified>
<Etag>0x8CAE7D0C4AF4487</Etag>
</Properties>
<Metadata>
<Color>orange</Color>
<ContainerNumber>01</ContainerNumber>
<SomeMetadataName>SomeMetadataValue</SomeMetadataName>
</Metadata>
</Container>
<Container>
<Name>container2</Name>
<Url>https://[url-for-service-account]/container2</Url>
<Properties>
<Last-Modified>Sun, 24 Sep 2018 17:26:40 GMT</Last-Modified>
<Etag>0x8CAE7CAD8C24928</Etag>
</Properties>
<Metadata>
<Color>pink</Color>
<ContainerNumber>02</ContainerNumber>
<SomeMetadataName>SomeMetadataValue</SomeMetadataName>
</Metadata>
```

```
</Container>
<Container>
<Name>container3</Name>
<Url>https://[url-for-service-account]/container3</Url>
<Properties>
<Last-Modified>Sun, 24 Sep 2018 17:26:40 GMT</Last-Modified>
<Etag>0x8CAE7CAD8EAC0BB</Etag>
</Properties>
<Metadata>
<Color>brown</Color>
<ContainerNumber>03</ContainerNumber>
<SomeMetadataName>SomeMetadataValue</SomeMetadataName>
</Metadata>
</Container>
</Containers>
<NextMarker>container4</NextMarker>
</EnumerationResults>
```

This approach requires a lot of manual parsing and the creation of HTTP packets to work with each API. For this reason, Azure provides pre-built *client libraries* that make working with the service easier for common languages and frameworks.

https://docs.microsoft.com/en-us/rest/api/storageservices/blob-service-rest-api

Question 48: Skipped

True or False: Azure Synapse functionality requires integration with Azure Data Factory, Azure Databricks and Power BI.

- True
- False **(Correct)**

Explanation
When thinking about usage patterns that customers are using today to maximize the value of their data, a modern data warehouse lets you bring together all your data at scale easily, so you get to the insights through analytics dashboards, operational reporting, or advanced analytics for your users.

The process of building a modern data warehouse typically consists of:

• Data Ingestion and Preparation.

• Making the data ready for consumption by analytical tools.

• Providing access to the data, in a shaped format so that it can easily be consumed by data visualization tools.

Prior to the release of Azure Synapse Analytics, this would be achieved in the following way.

Data ingestion and preparation

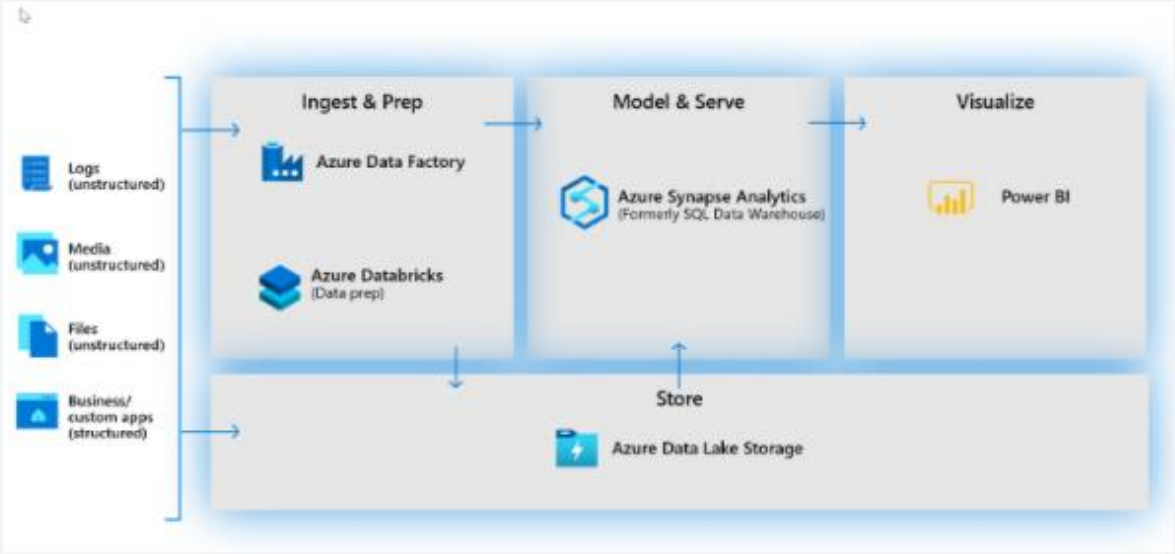
At the foundation, customers build a data lake to store all their data and different data types with Azure Data Lake Store Gen2.

To ingest data, customers can do so code-free with over 100 data integration connectors with Azure Data Factory. Data Factory empowers customers to do code-free ETL/ELT, including preparation and transformation.

And while a lot of our customers are currently heavily invested in the SQL Server Integration Services packages (SSIS), they created, they can leverage these without having to rewrite those packages in Azure Data Factory.

Whether the data is an on-premises data sources, other Azure services, or other cloud services, customers can seamlessly author, monitor, and manage their big data pipelines with a visual environment that is easy to use.

Another option for data preparation is Azure Databricks - to shape the data formats and prep it using a Notebook—making internal collaboration on data more streamlined and efficient.



Making the data ready for consumption by analytical tools

At the heart of a modern data warehouse, and cloud scale analytical solution was Azure Synapse Analytics (Formerly SQL Data Warehouse). This implemented a Massively Parallel Processing that brings together enterprise data warehousing and Big Data analytics.

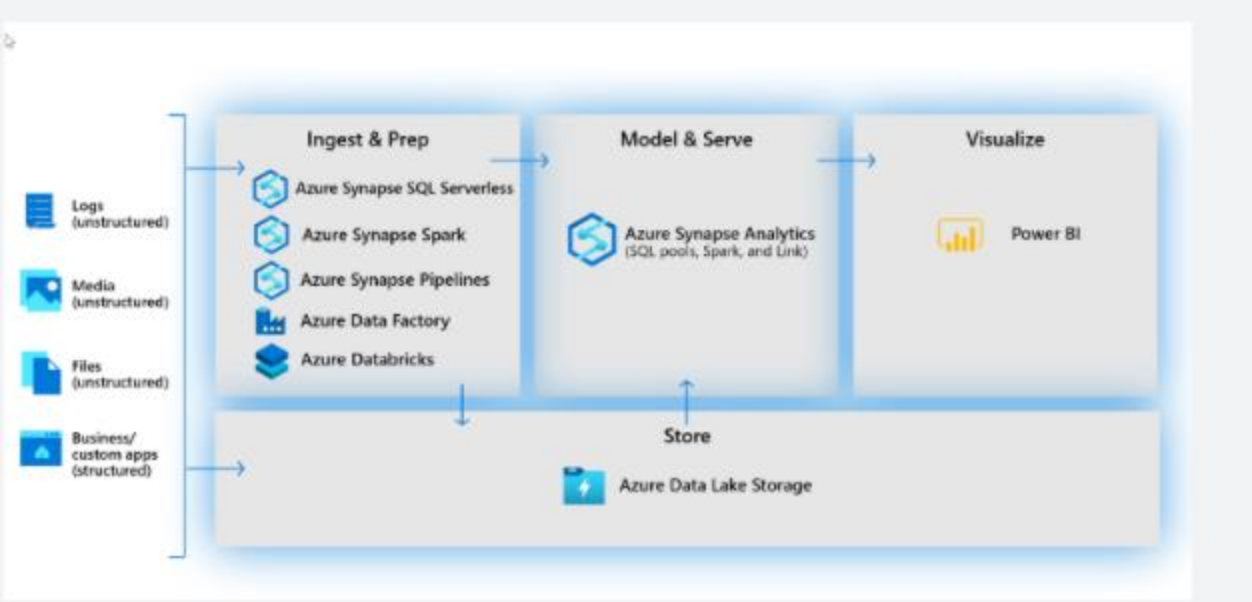
Providing access to the data, that it can easily be consumed by data visualization tools

Power BI enables customers to build visualizations on massive amounts of data and ensure that data insights are available to everyone across their organization.

Power BI supports an enormous set of data sources, which can be queried live, or be used to model and ingest, for detailed analysis and visualization.

Brought together with AI capabilities, it’s a powerful tool to build and deploy dashboards in the enterprise, through rich visualizations, and features like natural language querying.

With the release of Azure Synapse Analytics, you have a choice. You can either use Azure Synapse exclusively, which works very well for green field projects, but for organizations with existing investments in Azure with Azure Data Factory, Azure Databricks and Power BI, you can take a hybrid approach and combine them with Azure Synapse Analytics.



https://docs.microsoft.com/en-us/azure/synapse-analytics/overview-what-is

Question 49: Skipped

When creating a new cluster in the Azure Databricks workspace, what happens behind the scenes?

- Azure Databricks creates a cluster of driver and worker nodes, based on your VM type and size selections.
  (Correct)
- Azure Databricks provisions a dedicated VM that processes all jobs, based on your VM type and size selection.
- When an Azure Databricks workspace is deployed, you are allocated a pool of VMs. Creating a cluster draws from this pool.
- None of the listed options.

Explanation

At the time of cluster creation, you specify the types and sizes of the virtual machines (VMs) to use for both the Driver and Worker nodes, but Azure Databricks manages all other aspects of the cluster.

https://docs.microsoft.com/en-us/windows-server/remote/remote-desktop-services/virtual-machine-recs

Question 50: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

[?] is a Hadoop-compatible data repository that can store any size or type of data. The following are key features of [?]:

• Unlimited scalability

• Hadoop compatibility

• Security support for both access control lists (ACLs)

• POSIX compliance

• Zone-redundant storage

- Azure Data Lake Storage
  (Correct)
- Azure HDInsight
- Azure Cosmos DB
- Azure Data Studio
- Azure Bulk File Storage
- Azure Lab Services

Explanation
Azure Data Lake Storage is a Hadoop-compatible data repository that can store any size or type of data. This storage service is available as Generation 1 (Gen1) or Generation 2 (Gen2). Data Lake Storage Gen1 users don't have to upgrade to Gen2, but they forgo some benefits.

Data Lake Storage Gen2 users take advantage of Azure Blob storage, a hierarchical file system, and performance tuning that helps them process big-data analytics solutions. In Gen2, developers can access data through either the Blob API or the Data Lake file API. Gen2 can also act as a storage layer for a wide range of compute platforms, including Azure Databricks, Hadoop, and Azure HDInsight, but data doesn't need to be loaded into the platforms.

Here are the key features of Data Lake Storage:

• Unlimited scalability

• Hadoop compatibility

• Security support for both access control lists (ACLs)

• POSIX compliance

• An optimized Azure Blob File System (ABFS) driver that's designed for big-data analytics

• Zone-redundant storage

• Geo-redundant storage

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-introduction

Question 51: Skipped

Scenario: O'Shaughnessy's is a fast food restaurant. The chain has stores nationwide and is rivalled by Big Belly Burgers. You have been hired by the company to advise on the implementation of Azure migrating from an on-prem datacentre.

The IT team has an Azure subscription which contains an Azure Storage account and they plan to create an Azure container instance named OShaughnessy001 that will use a Docker image named Source001. Source001 contains a Microsoft SQL Server instance that requires persistent storage. Right now the team is configuring a storage service for OShaughnessy001 and there is debate around which of the following should be used.

As the expert consultant, the team looks to you for direction.

Which should you advise them to use?

- Azure Blob storage
- Azure Files
  (Correct)
- Azure Table storage
- Azure Queue storage

Explanation

Persistent Docker volumes with Azure File Storage

Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol or Network File System (NFS) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments. Azure Files SMB file shares are accessible from Windows, Linux, and macOS clients. Azure Files NFS file shares are accessible from Linux or macOS clients. Additionally, Azure Files SMB file shares can be cached on Windows Servers with Azure File Sync for fast access near where the data is being used.

Azure file shares can be used to:

Replace or supplement on-premises file servers:
Azure Files can be used to completely replace or supplement traditional on-premises file servers or NAS devices. Popular operating systems such as Windows, macOS, and Linux can directly mount Azure file shares wherever they are in the world. Azure File SMB file shares can also be replicated with Azure File Sync to Windows Servers, either on-premises or in the cloud, for performance and distributed caching of the data where it's being used. With the recent release of Azure Files AD Authentication, Azure File SMB file shares can continue to work with AD hosted on-premises for access control.

"Lift and shift" applications:
Azure Files makes it easy to "lift and shift" applications to the cloud that expect a file share to store file application or user data. Azure Files enables both the "classic" lift and shift scenario, where both the application and its data are moved to Azure, and the "hybrid" lift and shift scenario, where the application data is moved to Azure Files, and the application continues to run on-premises.

Simplify cloud development:
Azure Files can also be used in numerous ways to simplify new cloud development projects. For example:

Shared application settings:
A common pattern for distributed applications is to have configuration files in a centralized location where they can be accessed from many application instances. Application instances can load their configuration through the File REST API, and humans can access them as needed by mounting the SMB share locally.

Diagnostic share:
An Azure file share is a convenient place for cloud applications to write their logs, metrics, and crash dumps. Logs can be written by the application instances via the File REST API, and developers can access them by mounting the file share on their local machine. This enables great flexibility, as developers can embrace cloud development without having to abandon any existing tooling they know and love.

Dev/Test/Debug:
When developers or administrators are working on VMs in the cloud, they often need a set of tools or utilities. Copying such utilities and tools to each VM can be a time consuming exercise. By mounting an Azure file share locally on the VMs, a developer and administrator can quickly access their tools and utilities, no copying required.

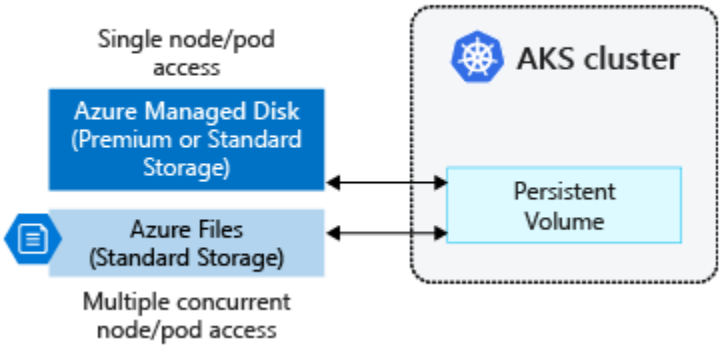
Containerization:
Azure file shares can be used as persistent volumes for stateful containers. Containers deliver "build once, run anywhere" capabilities that enable developers to accelerate innovation. For the containers that access raw data at every start, a shared file system is required to allow these containers to access the file system no matter which instance they run on.

https://docs.microsoft.com/en-us/azure/storage/files/storage-files-introduction

Persistent volumes

Volumes defined and created as part of the pod lifecycle only exist until you delete the pod. Pods often expect their storage to remain if a pod is rescheduled on a different host during a maintenance event, especially in StatefulSets. A *persistent volume* (PV) is a storage resource created and managed by the Kubernetes API that can exist beyond the lifetime of an individual pod.

You can use Azure Disks or Files to provide the PersistentVolume. As noted in the Volumes section, the choice of Disks or Files is often determined by the need for concurrent access to the data or the performance tier.

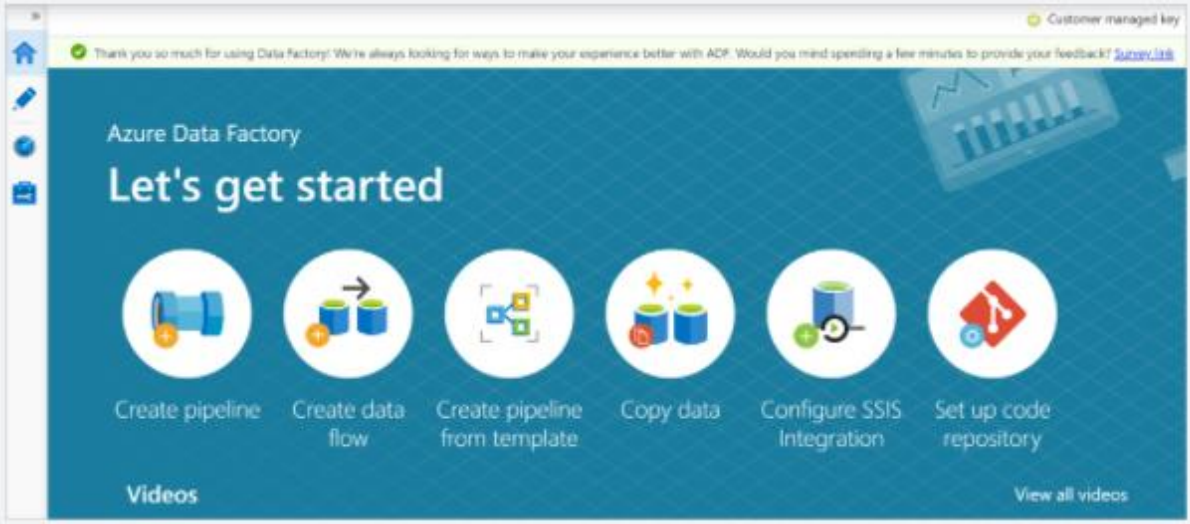


A PersistentVolume can be *statically* created by a cluster administrator, or *dynamically* created by the Kubernetes API server. If a pod is scheduled and requests currently unavailable storage, Kubernetes can create the underlying Azure Disk or Files storage and attach it to the pod. Dynamic provisioning uses a *StorageClass* to identify what type of Azure storage needs to be created.

https://docs.microsoft.com/en-us/azure/aks/concepts-storage#volumes

Question 52: Skipped

Scenario: You team is working on a project using the Azure Data Factory authoring tool.



A junior team member comes to you and asks *"Where can I find the Copy Data activity ?"*

Which of the below is the correct location?

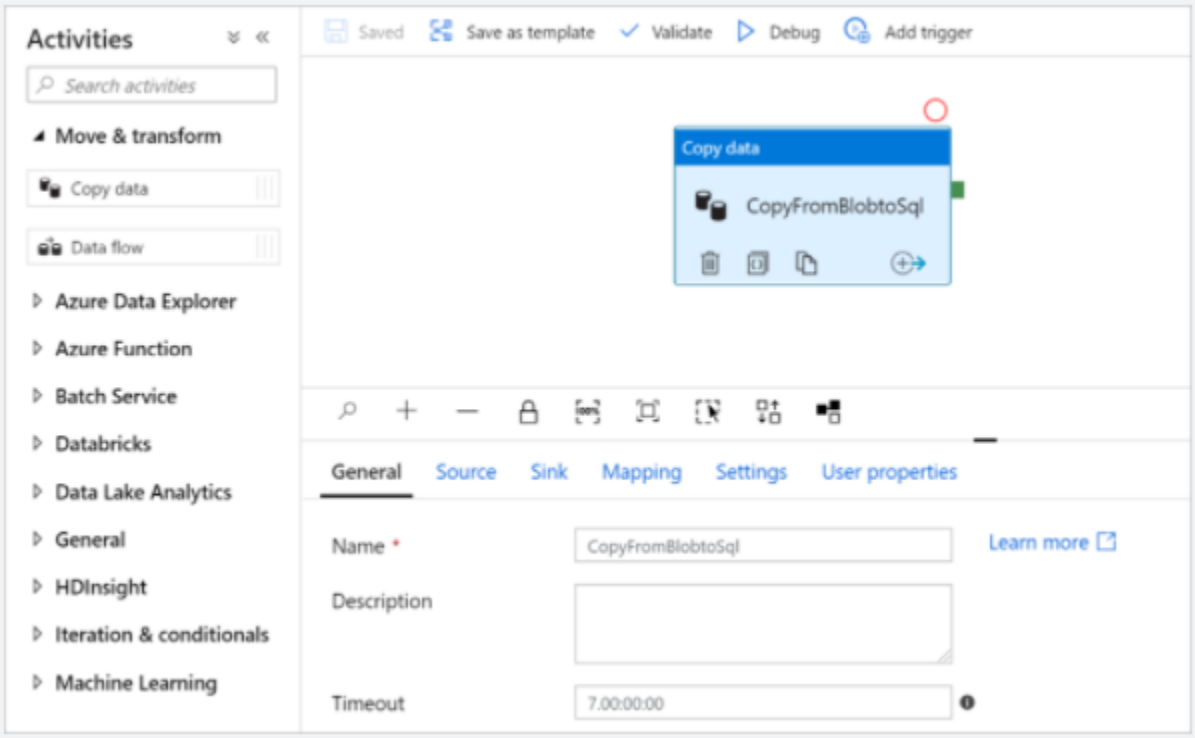
- Data Explorer
- Azure Function
- Batch Service
- Batch Service
- Move & Transform

(Correct)

- Databricks

Explanation
The Move & Transform section contains activities that are specific to Azure Data Factory copying data and defining data flows.

Activities tool box → **Move and Transform** category → **Copy Data** activity



https://docs.microsoft.com/en-us/azure/data-factory/author-visually

Question 53: Skipped

Scenario: You are working as a consultant at Avengers Security (AS). At the moment, you are consulting with Tony, the lead of the IT team and the topic of discussion is about creating an SQL pool in Azure Synapse that will use data from the data lake.

AS has an enterprise-wide Azure Data Lake Storage Gen2 account where the data lake is accessible only through an Azure virtual network named VNET1. The AS sales team members are in an Azure Active Directory group named Sales. POSIX controls are used to assign the Sales group access to the files in the data lake. The plan is to load data to the SQL pool every hour and the team needs to ensure that the SQL pool can load the sales data from the data lake.

Since Azure is new to Avengers Security, the team has created the list shown below of things the team members think are needed, but they are not certain about which actions are necessary, and which are not.

Tony and the team look to you for guidance; which of the following should you advise them to perform? (Select three)

- ☐ Add the managed identity to the Sales group.
  (Correct)
- ☐ Add your Azure Active Directory (Azure AD) account to the Sales group.
- ☐ Create a managed identity.
  (Correct)
- ☐ Use the snared access signature (SAS) as the credentials for the data load process.
- ☐ Create a shared access signature (SAS).
- ☐ Use the managed identity as the credentials for the data load process.
  (Correct)

Explanation
You advise them to perform the following:

• *Create a managed identity.*

• *Add the managed identity to the Sales group.*

• Use the managed identity as the credentials for the data load process.

Azure Data Lake Storage Gen2 implements an access control model that supports both Azure role-based access control (Azure RBAC) and POSIX-like access control lists (ACLs). This article describes access control lists in Data Lake Storage Gen2. To learn about how to incorporate Azure RBAC together with ACLs, and how system evaluates them to make authorization decisions, see Access control model in Azure Data Lake Storage Gen2.

About ACLs

You can associate a security principal with an access level for files and directories. These associations are captured in an *access control list (ACL)*. Each file and directory

in your storage account has an access control list. When a security principal attempts an operation on a file or directory, An ACL check determines whether that security principal (user, group, service principal, or managed identity) has the correct permission level to perform the operation

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-access-control

You can use Storage Explorer to view, and then update the ACLs of directories and files. ACL inheritance is already available for new child items that are created under a parent directory. But you can also apply ACL settings recursively on the existing child items of a parent directory without having to make these changes individually for each child item.

Storage Explorer makes use of both the Blob (blob) & Data Lake Storage Gen2 (dfs) endpoints when working with Azure Data Lake Storage Gen2. If access to Azure Data Lake Storage Gen2 is configured using private endpoints, ensure that two private endpoints are created for the storage account: one with the target sub-resource `blob` and the other with the target sub-resource `dfs`.

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-explorer-acl

Question 54: Skipped

Knowing now the different concepts of spark it is imperative to understand how it fits in with the different Data services on Azure.

Which of the following is best described by:

"Organizations that don't have existing spark implementations yet, get the functionality to spin up a spark cluster to meet data engineering needs without the overhead of the other Spark Platforms. Data Engineers, Data scientist, Data Platform Experts, and Data Analyst can come together within Synapse Analytics where the Spark cluster is spun up quickly to meet the needs. It provides scale in an efficient way for Spark Clusters and integrates with the one stop shop Data warehousing platform of Synapse."

- Apache Spark
- HDI
- Azure Databricks

- Spark Pools in Azure Synapse Analytics
  (Correct)

Explanation
There are two concepts within Apache Spark Pools in Azure Synapse Analytics, namely Spark pools and Spark Instances. In short, they do the following:

Spark Pools:

• Exists as Metadata

• Creates a Spark Instance

• No costs associated with creating Pool

• Permissions can be applied

• Best practices

Spark Instances:

• Created when connected to Spark Pool, Session, or Job

• Multiple users can have access

• Reusable

Knowing now the different concepts of spark it is imperative to understand how it fits in with the different Data services on Azure. Below is a table where "the when to use what" is outlined:

	Apache Spark	HDInsight	Azure Databricks	Synapse Spark
What	Is an Open Source memory optimized system for managing big data workloads	Microsoft implementation of Open Source Spark managed within the realms of Azure	AA managed Spark as a Service solution	Embedded Spark capability within Azure Synapse Analytics
When	When you want to benefits of spark for big data processing and/or data science work without the Service Level Agreements of a provider	When you want to benefits of OSS spark with the Service Level Agreement of a provide	Provides end to end data engineering and data science solution and management platform	Enables organizations without existing Spark implementations to fire up a Spark cluster to meet data engineering needs without the overheads of the other Spark platforms listed
Who	Open Source Professionals	Open Source Professionals wanting SLA's and Microsoft Data Platform experts	Data Engineers and Data Scientists working on big data projects every day	Data Engineers, Data Scientists, Data Platform experts and Data Analysts
Why	To overcome the limitations of SMP systems imposed on big data workloads	To take advantage of the OSS Big Data Analytics platform with SLA's in place to ensure business continuity	It provides the ability to create and manage an end to end big data/data science project using one platform	It provides the ability to scale efficiently with spark clusters within a one stop shop DataWarehousing platform of Synapse.

Spark Pools in Azure Synapse Analytics: Spark in Azure Synapse Analytics is a capability of Spark embedded in Azure Synapse Analytics in which organizations that don't have existing spark implementations yet, get the functionality to spin up a spark cluster to meet data engineering needs without the overhead of the other Spark Platforms listed. Data Engineers, Data scientist, Data Platform Experts, and Data Analyst can come together within Synapse Analytics where the Spark cluster is spun up quickly to meet the needs. It provides scale in an efficient way for Spark Clusters and integrates with the one stop shop Data warehousing platform of Synapse.

Apache Spark: Apache Spark is an open-source memory optimized system for managing big data workloads, which is used when you want a spark engine for big data processing or data science where you don't mind that there is no SLA provided. Usually it is of interest of Open Source Professionals and the reason for Apache spark is to overcome the limitations of what was known as SMP systems for big data workloads.

HDI: HDI is an implementation by Microsoft of Open Source Spark, managed on the Azure Platform. You can use HDI for a spark environment when you are aware of the benefits of Apache Spark in its OSS form, but you want a SLA. Usually this of interest of Open Source Professionals needing an SLA as well as Data Platform experts experienced with Microsoft.

Azure Databricks: Azure Databricks is a managed Spark as a Service propriety Solution that provides an end to end data engineering/data science platform as a solution. Azure Databricks is of interest for Data Engineers and Data Scientists, working on big data projects daily because it provides the whole platform in which you have the ability to create and manage the big data/data science pipelines/projects all on one platform.

https://docs.microsoft.com/en-us/azure/synapse-analytics/spark/apache-spark-overview

Question 55: Skipped

The first step in deploying Azure Synapse Analytics is to deploy an Azure Synapse Analytics workspace. A shared Hive-compatible metadata system allows tables defined on files in the data lake to be seamlessly consumed by either Spark or Hive.

SQL and Spark can directly explore and analyze which types of files stored in the data lake? (Select all that apply)

- ☐ CSV **(Correct)**
- ☐ Parquet **(Correct)**
- ☐ XLSX
- ☐ TXT
- ☐ JSON **(Correct)**

- XLS
- TSV
  (Correct)
- PDF

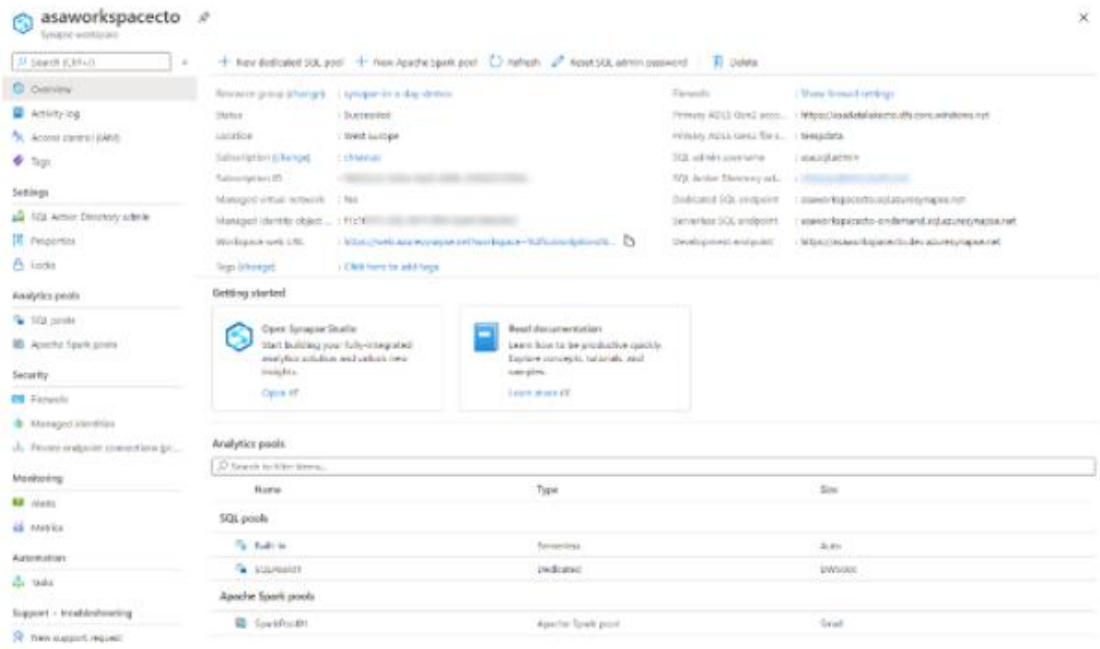
Explanation
The first step in deploying Azure Synapse Analytics is to deploy an Azure Synapse Analytics workspace. This deployment creates several resources which include an Azure Data Lake Storage Gen2 account that acts as the primary storage and the container to store workspace data. The workspace stores data in Apache Spark tables. It also stores Spark application logs under a folder called `/synapse/workspacename`. There are endpoints created that can be used to connect to the SQL on-demand service, and the Azure Synapse Analytics Workspace itself.

Azure Synapse Analytics enables you to create pools, either SQL pools, or Spark pools within the workspace that can be seamlessly mixed and matched based on your requirements. It is able to do this through Azure Synapse Analytics shared metadata, which enables the different engines to share databases and tables.

For example, **A shared Hive-compatible metadata system allows tables defined on files in the data lake to be seamlessly consumed by either Spark or Hive. SQL and Spark can directly explore and analyze Parquet, CSV, TSV, and JSON files stored in the data lake.** There is also a fast scalable load and unload for data going between SQL and Spark databases.

It is this capability that enables the Modern Data Warehousing workload pattern and gives the workspace SQL engines access to databases and tables created with Spark. It also allows the SQL engines to create their own objects that aren't being shared with the other engines.

The Azure Synapse Analytics workspace is the central location where you can view information about these resources and connect to them from within the Azure portal. The initial setup looks as follows:



With a SQL on-demand endpoint available, and an Azure Data Lake Storage Gen2 (ADLS Gen2) account, you can immediately realize value from the product by uploading files to the data lake, and using the SQL on-demand service to prepare and explore the files

Furthermore, while you are able to manage some aspects of the service in the Azure portal, the best practices is to connect to the Azure Synapse Studio to perform your activities from with there.

https://docs.microsoft.com/en-us/azure/synapse-analytics/quickstart-create-workspace

Question 56: Skipped

Which Index Type offers the highest compression in Synapse Analytics?

- Replicated
- Columnstore
  (Correct)
- Heap
- Round-Robin

- Rowstore

Explanation
Columnstore is the default index type created for a table. It works on segments of rows that get compressed and optimized by column.

Dedicated SQL Pools have the following indexing options available:

Clustered columnstore index

Dedicated SQL Pools create a clustered columnstore index when no index options are specified on a table. Clustered columnstore indexes offer both the highest level of data compression as well as the best overall query performance. Clustered columnstore indexes will generally outperform clustered rowstore indexes or heap tables and are usually the best choice for large tables.

Additional compression on the data can be gained also with the index option `COLUMNSTORE_ARCHIVE`. These reduced sizes allow less memory to be used when accessing and using the data as well as reducing the IOPs required to retrieve data from storage.

Columnstore works on segments of 1,024,000 rows that get compressed and optimized by column. This segmentation further helps to filter out and reduce the data accessed through leveraging metadata stored which summarizes the range and values within each segment during query optimization.

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/sql-data-warehouse-tables-index

Clustered index

Clustered Rowstore Indexes define how the table itself is stored, ordered by the columns used for the Index. There can be only one clustered index on a table.

Clustered indexes are best for queries and joins that require ranges of data to be scanned, preferably in the same order that the index is defined.

Non-clustered index

A non-clustered index can be defined on a table or view with a clustered index or on a heap. Each index row in the non-clustered index contains the non-clustered key value

and a row locator. This is a data structure separate/additional to the table or heap. You can create multiple non-clustered indexes on a table.

Non clustered indexes are best used when used for the columns in a join, group by statement or where clauses that return an exact match or few rows.

https://docs.microsoft.com/en-us/sql/t-sql/statements/create-table-azure-sql-data-warehouse?view=aps-pdw-2016-au7

Question 57: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

Azure Synapse Analytics is an integrated analytics platform, which combines data warehousing, big data analytics, data integration, and visualization into a single environment. Azure Synapse Analytics empowers users of all abilities to gain access and quick insights across all of their data, enabling a whole new level of performance and scale.

Diagnostic analytics deals with answering the question [?].

- *"Why is it happening?"*
  (Correct)
- *"What is likely to happen in the future based on previous trends and patterns?"*
- *"When will the modification made meet my goals?"*
- *"What is happening in my business?"*

Explanation
Azure Synapse Analytics is an integrated analytics platform, which combines data warehousing, big data analytics, data integration, and visualization into a single environment. Azure Synapse Analytics empowers users of all abilities to gain access and quick insights across all of their data, enabling a whole new level of performance and scale.

Gartner defines a range of analytical types that Azure Synapse Analytics can support including:

Descriptive analytics

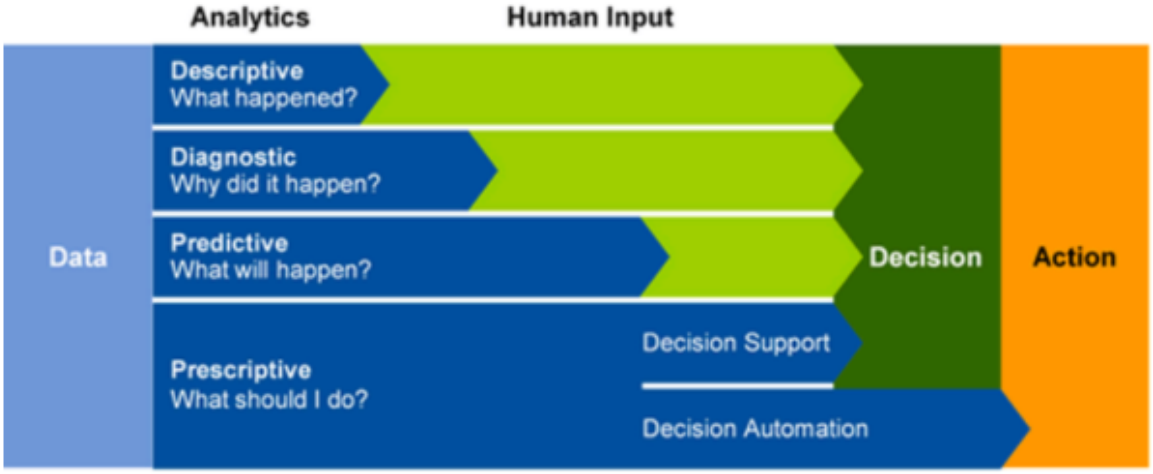
Descriptive analytics answers the question “What is happening in my business?” The data to answer this question is typically answered through the creation of a data warehouse. Azure Synapse Analytics leverages the dedicated SQL Pool capability that enables you to create a persisted data warehouse to perform this type of analysis. You can also make use of SQL Serverless to prepare data from files to create a data warehouse interactively to answer the question too.

Diagnostic analytics

Diagnostic analytics deals with answering the question **“Why is it happening?”** this may involve exploring information that already exists in a data warehouse, but typically involves a wider search of your data estate to find more data to support this type of analysis.

You can use the same SQL serverless capability within Azure Synapse Analytics that enables you to interactively explore data within a data lake. This can quickly enable a user to search for additional data that may help them to understand “Why is it happening?”

https://www.valamis.com/hub/descriptive-analytics



Predictive analytics

Azure Synapse Analytics also enables you to answer the question “What is likely to happen in the future based on previous trends and patterns?” by using its integrated Apache Spark engine. This can also be used in conjunction with other services such as Azure Machine Learning Services, or Azure Databricks.

https://www.ibm.com/analytics/predictive-analytics

Prescriptive analytics

This type of analytics looks at executing actions based on real-time or near real-time analysis of data, using predictive analytics. Azure Synapse Analytics provides this capability through both Apache Spark, Azure Synapse Link, and by integrating streaming technologies such as Azure Stream Analytics.

https://www.talend.com/resources/what-is-prescriptive-analytics/

Azure Synapse Analytics gives the users of the service the freedom to query data on their own terms, using either serverless or dedicated resources at scale. Azure Synapse Analytics brings these two worlds together with a unified data integration experience to ingest, prepare, manage, and serve data using Azure Synapse Pipelines. In addition, you can visualize the data in the form of dashboards and reports for immediate analysis using Power BI which is integrated into the service too.

https://docs.microsoft.com/en-us/azure/synapse-analytics/overview-what-is

Question 58: Skipped

Scenario: Your organization must respond to data events in real time in a continuous time-bound stream. The company must monitor IoT devices combined with remote patient monitoring to dispatch life-critical services.

Which would be the best Azure product to use?

- Azure Table Storage
- Azure Cosmos DB
- Azure On-prem solution
- Azure DataNow
- Azure Synapse Analytics
- Azure Stream Analytics

(Correct)

Explanation
Applications, sensors, monitoring devices, and gateways broadcast continuous event data known as *data streams*. Streaming data is high volume and has a lighter payload than nonstreaming systems.

Data engineers use Azure Stream Analytics to process streaming data and respond to data anomalies in real time. You can use Stream Analytics for Internet of Things (IoT) monitoring, web logs, remote patient monitoring, and point of sale (POS) systems.

When to use Stream Analytics

If your organization must respond to data events in real time or analyze large batches of data in a continuous time-bound stream, Stream Analytics is a good solution. Your organization must decide whether to work with streaming data or batch data.

In real time, data is ingested from applications or IoT devices and gateways into an event hub or IoT hub. The event hub or IoT hub then streams the data into Stream Analytics for real-time analysis.

Batch systems process groups of data that are stored in an Azure Blob store. They do this in a single job that runs at a predefined interval. Don't use batch systems for business intelligence systems that can't tolerate the predefined interval. For example, an autonomous vehicle can't wait for a batch system to adjust its driving. Similarly, a fraud-detection system must decline a questionable financial transaction in real time.

Data ingestion

As a data engineer, set up data ingestion in Stream Analytics by configuring data inputs from first-class integration sources. These sources include Azure Event Hubs, Azure IoT Hub, and Azure Blob storage.

An IoT hub is the cloud gateway that connects IoT devices. IoT hubs gather data to drive business insights and automation.

Features in Azure IoT Hub enrich the relationship between your devices and your back-end systems. Bidirectional communication capabilities mean that while you receive data from devices, you can also send commands and policies back to devices. Take advantage of this ability, for example, to update properties or invoke device management actions. Azure IoT Hub can also authenticate access between the IoT device and the IoT hub.

Azure Event Hubs provides big-data streaming services. It's designed for high data throughput, allowing customers to send billions of requests per day. Event Hubs uses a

partitioned consumer model to scale out your data stream. This service is integrated into the big-data and analytics services of Azure. These include Databricks, Stream Analytics, Azure Data Lake Storage, and HDInsight. Event Hubs provides authentication through a shared key.

You can use Azure Storage to store data before you process it in batches.

IoT Capability	IoT Hub standard tier	IoT Hub basic tier	Event Hubs
Device-to-cloud messaging	✓	✓	✓
Protocols: HTTPS, AMQP, AMQP over webSockets	✓	✓	✓
Protocols: MQTT, MQTT over webSockets	✓	✓	
Per-device identity	✓	✓	
File upload from devices	✓	✓	
Device Provisioning Service	✓	✓	
Cloud-to-device messaging	✓		
Device twin and device management	✓		
IoT Edge	✓		

https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-introduction

Question 59: Skipped

Scenario: Jungle.com uses Azure Cosmos DB to store sales orders and customer profile data from their eCommerce site. The NoSQL document store provided by the Azure Cosmos DB provides the familiarity of managing their data using SQL syntax, while being able to read and write the files at a massive, global scale.

While Jungle.com is happy with the capabilities and performance of Azure Cosmos DB, they are concerned about the cost of executing a large volume of complex analytical queries needed to fulfill their operational reporting requirements.

They want to efficiently access all their operational data stored in Cosmos DB without needing to increase the Azure Cosmos DB throughput and associate cost. They have

looked at options for extracting data from their containers to the data lake as it changes, through the Azure Cosmos DB change feed mechanism.

The problem with this approach is the extra service and code dependencies and long-term maintenance of the solution. They could perform bulk exports from a Synapse Pipeline, but then they won't have the most up-to-date information at any given moment.

Which would be the best action to take?

- Enable Azure VNet Peering for Cosmos DB and enable the analytical store on their Azure Cosmos DB containers.
- Enable Azure Dedicated Connect for Cosmos DB and enable the analytical store on their Azure Cosmos DB containers.
- Enable Azure Synapse Link for Cosmos DB and enable the analytical store on their Azure Cosmos DB containers.
  (Correct)
- Enable Azure VPN Gateway for Cosmos DB and enable the analytical store on their Azure Cosmos DB containers.

Explanation

The best option is to enable Azure Synapse Link for Cosmos DB and enable the analytical store on their Azure Cosmos DB containers. With this configuration, all transactional data is automatically stored in a fully isolated column store. This store enables large-scale analytics against the operational data in Azure Cosmos DB, without impacting the transactional workloads or incurring additional costs. Azure Synapse Link for Cosmos DB creates a tight integration between Azure Cosmos DB and Azure Synapse Analytics, which enables Jungle.com to run near real-time analytics over their operational data with no-ETL and full performance isolation from their transactional workloads.

Hybrid Transactional/Analytics Processing (HTAP) enables businesses to perform analytics over database systems that provide high performance transactional capabilities without impacting the performance of these systems. This enables organizations to use a database to fulfill both transactional and analytical needs to support near real-time analysis of operational data to make timely decisions.

By combining the distributed scale of Cosmos DB's transactional processing and the built-in analytical store with the computing power of Azure Synapse Analytics, Azure Synapse Link enables a Hybrid Transactional/Analytical Processing (HTAP) architecture for optimizing Jungle.com business processes. This integration eliminates ETL processes, enabling business analysts, data engineers and data scientists to self-serve

and run near real-time BI, analytics, and Machine Learning pipelines over operational data.

https://azure.microsoft.com/en-us/services/synapse-analytics/

Question 60: Skipped

True or False: The Apache Spark history server can be used to debug and diagnose completed only.

- True
- False **(Correct)**

Explanation
The Apache Spark history server can be used to debug and diagnose completed as well as running Spark applications.

You can use the Apache Spark history server web UI from the Azure Synapse Studio environment. Once you launch it, there are several tabs that you can use in order to monitor the Apache Spark application:

• Jobs

• Stages

• Storage

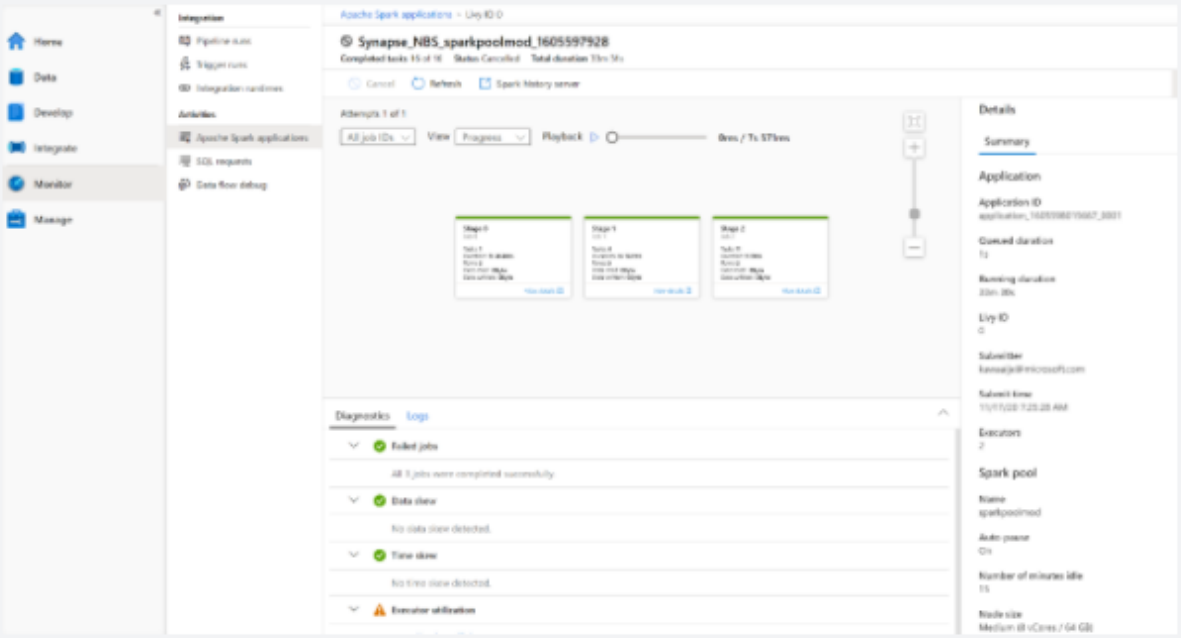
• Environment

• Executors

• SQL

The Apache Spark history server is the web user interface known as Spark UI for completed and running Spark applications. If you want to navigate to the Apache Spark History server, you can navigate to the Azure Synapse Analytics studio environment and go to the Monitor tab. In the Monitor tab, you can select 'Apache Spark Applications'.

To give you a visual interpretation of how that looks like, see below:



If you are familiar with Apache Spark, you can find the standard Apache Spark history server UI by selecting Open Spark UI.

The other possibility of opening the Apache Spark History server is by navigating to the Data tab, where if you create a notebook and read a dataframe you can go to the bottom of the page where you'll find the Spark History Server known as the Spark UI.

https://docs.microsoft.com/en-us/azure/synapse-analytics/spark/apache-spark-history-server

Question 61: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

Microsoft Azure Storage is a managed service that provides durable, secure, and scalable storage in the cloud. Azure Files enables you to set up highly available network file shares that can be accessed using the standard Server Message Block (SMB) protocol. This means that multiple VMs can share the same files with both read and write access. You can read the files using the REST interface or the storage client libraries. You can also associate a unique URL to any file to allow fine-grained access to a private file for a set period of time.

Which are common scenarios where File shares can be used? (Select all that apply)

- ☐ Storing shared configuration files for VMs, tools, or utilities so that everyone is using unique versions.
- ☐ Shared data between on-premises applications and Azure VMs to allow migration of apps to the cloud over a period of time.
(Correct)
- ☐ Shared data between on-premises applications and Azure VMs to allow migration of apps to the cloud instantly.
- ☐ Log files such as diagnostics, metrics, and crash dumps.
(Correct)
- ☐ Storing shared configuration files for VMs, tools, or utilities so that everyone is using the same version.
(Correct)

Explanation
Microsoft Azure Storage is a managed service that provides durable, secure, and scalable storage in the cloud.

Durable	Redundancy ensures that your data is safe in the event of transient hardware failures. You can also replicate data across datacenters or geographical regions for extra protection from local catastrophe or natural disaster. Data replicated in this way remains highly available in the event of an unexpected outage.
Secure	All data written to Azure Storage is encrypted by the service. Azure Storage provides you with fine-grained control over who has access to your data.
Scalable	Azure Storage is designed to be massively scalable to meet the data storage and performance needs of today's applications.
Managed	Microsoft Azure handles maintenance and any critical problems for you.

A single Azure subscription can host up to 200 storage accounts, each of which can hold 500 TB of data.

Azure data services

Azure storage includes four types of data:

• Azure Blobs: A massively scalable object store for text and binary data. Can include support for Azure Data Lake Storage Gen2.

• **Files**: Managed file shares for cloud or on-premises deployments.

• Azure Queues: A messaging store for reliable messaging between application components.

• Azure Tables: A NoSQL store for schema-less storage of structured data. Table Storage is not covered in this module.

• Azure Disks: Block-level storage volumes for Azure VMs.

All of these data types in Azure Storage are accessible from anywhere in the world over HTTP or HTTPS. Microsoft provides SDKs for Azure Storage in various languages, and a REST API. You can also visually explore your data right in the Azure portal.

Files

Azure Files enables you to set up highly available network file shares that can be accessed using the standard Server Message Block (SMB) protocol. This means that multiple VMs can share the same files with both read and write access. You can also read the files using the REST interface or the storage client libraries. You can also associate a unique URL to any file to allow fine-grained access to a private file for a set period of time. File shares can be used for many common scenarios:

• Storing shared configuration files for VMs, tools, or utilities so that everyone is using the same version.

• Log files such as diagnostics, metrics, and crash dumps.

• Shared data between on-premises applications and Azure VMs to allow migration of apps to the cloud over a period of time.

https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction

Question 62: Skipped

When talking about the Azure Databricks workspace, we refer to two different things.

• The first reference is the logical Azure Databricks environment in which clusters are created, data is stored (via DBFS), and in which the server resources are housed.

• The second reference is the more common one used within the context of Azure Databricks.

The first step to using Azure Databricks is to create and deploy a Databricks workspace, which is the logical environment. You can do this in the Azure portal.

There are a number of required values to create your Azure Databricks workspace.

Which are they? (Select five)

- ☐ Pricing Tier **(Correct)**
- ☐ Subscription **(Correct)**
- ☐ Autopilot Options
- ☐ Cluster Mode
- ☐ Resource Group **(Correct)**
- ☐ Workspace Name **(Correct)**
- ☐ Databricks RuntimeVersion
- ☐ Location **(Correct)**
- ☐ Node Type

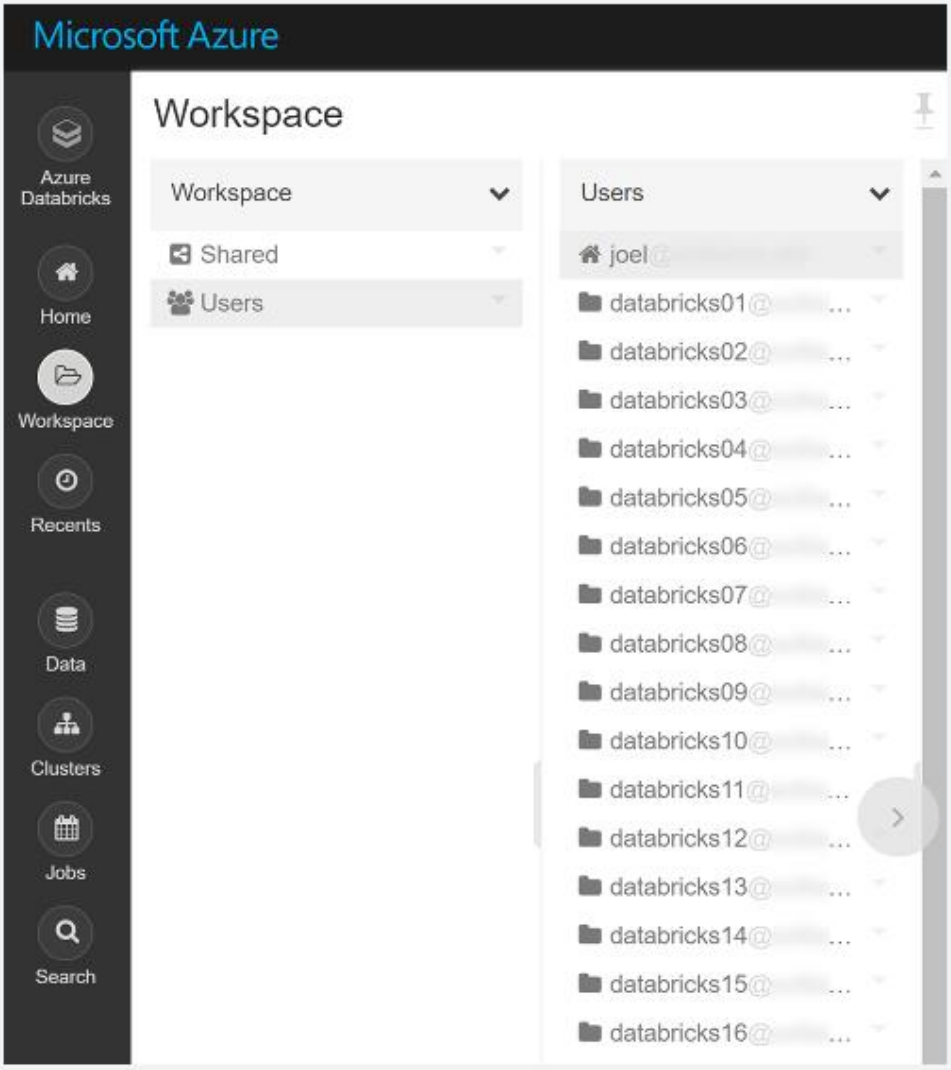
Explanation

When talking about the Azure Databricks workspace, we refer to two different things.

• The first reference is the logical Azure Databricks environment in which clusters are created, data is stored (via DBFS), and in which the server resources are housed.

• The second reference is the more common one used within the context of Azure Databricks.

That is the special root folder for all of your organization's Databricks assets, including notebooks, libraries, and dashboards, as shown below:



The first step to using Azure Databricks is to create and deploy a Databricks workspace, which is the logical environment. You can do this in the Azure portal.

Deploy an Azure Databricks workspace

1. Open the Azure portal.

2. Click **Create a Resource** in the top left

3. Search for "Databricks"

4. Select *Azure Databricks*

5. On the Azure Databricks page select *Create*

6. Provide the required values to create your Azure Databricks workspace:

- **Subscription**: Choose the Azure subscription in which to deploy the workspace.

- **Resource Group**: Use **Create new** and provide a name for the new resource group.

- **Location**: Select a location near you for deployment. For the list of regions that are supported by Azure Databricks, see Azure services available by region.

- **Workspace Name**: Provide a unique name for your workspace.

- **Pricing Tier**: **Trial (Premium - 14 days Free DBUs)**. You must select this option when creating your workspace or you will be charged. The workspace will suspend automatically after 14 days. When the trial is over you can convert the workspace to **Premium** but then you will be charged for your usage.

7.Select **Review + Create**.

8.Select **Create**.

The workspace creation takes a few minutes. During workspace creation, the **Submitting deployment for Azure Databricks** tile appears on the right side of the portal. You might need to scroll right on your dashboard to see the tile. There's also a progress bar displayed near the top of the screen. You can watch either area for progress.

What is a cluster?

The notebooks are backed by clusters, or networked computers, that work together to process your data. The first step is to create a cluster.

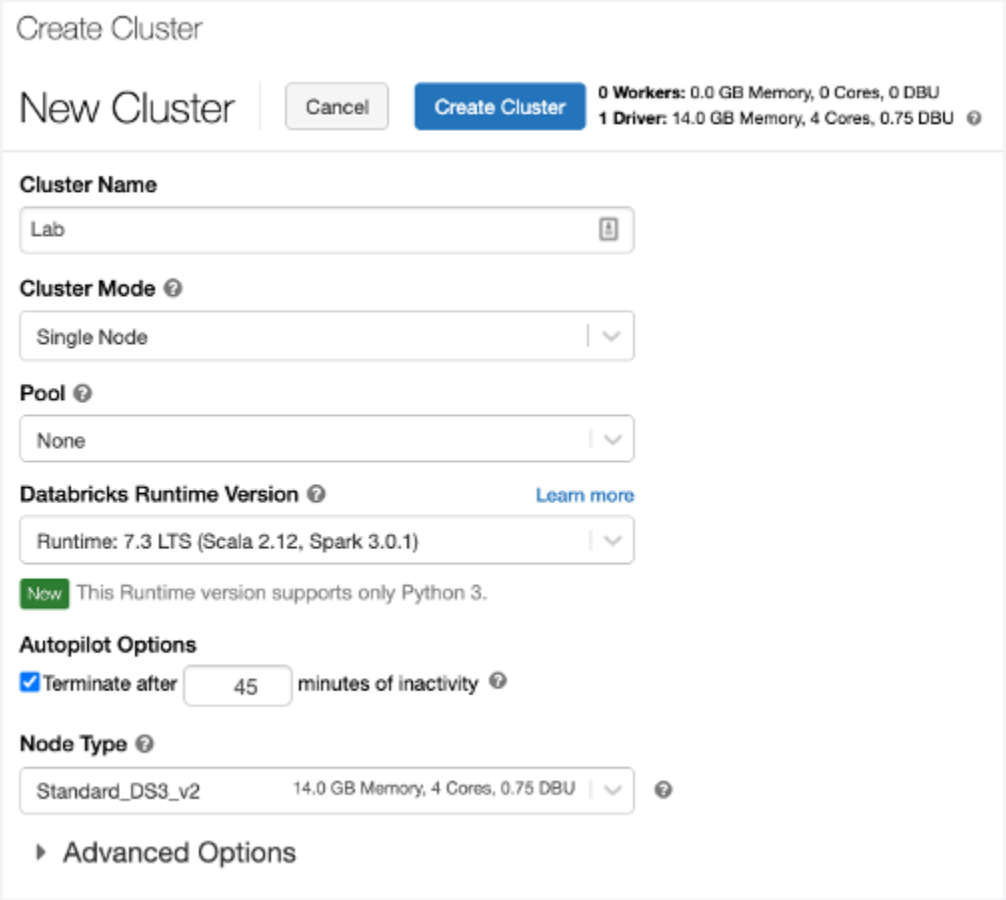
Create a cluster

1. When your Azure Databricks workspace creation is complete, select the link to go to the resource.

2. Select **Launch Workspace** to open your Databricks workspace in a new tab.

3. In the left-hand menu of your Databricks workspace, select **Clusters**.

4. Select **Create Cluster** to add a new cluster.



5. Enter a name for your cluster. Use your name or initials to easily differentiate your cluster from your coworkers.

6. Select the Cluster Mode: Single Node.

7. Select the Databricks RuntimeVersion: Runtime: 7.3 LTS (Scala 2.12, Spark 3.0.1).

8. Under Autopilot Options, leave the box checked and in the text box enter 45.

9. Select the Node Type: Standard_DS3_v2.

10. Select Create Cluster.

https://docs.microsoft.com/en-us/azure/databricks/clusters/create

Question 63: Skipped

You can use either the REST API or the Azure client library to programmatically access a storage account. What is the primary advantage of using the client library?

- Cost
- Availability
- Convenience **(Correct)**
- Localization

Explanation
Code that uses the client library is much shorter and simpler than code that uses the REST API. The client library handles assembling requests and parsing responses for you.

https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

Question 64: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

A(n) [?] is an orchestration of pipeline activities that includes chaining activities in a sequence, branching, defining parameters at the pipeline level, and passing arguments while invoking the pipeline on demand or from a trigger.

- Data Flow
- Control Flow **(Correct)**
- Procedure
- Workflow
- Test Lab
-

Activity

Explanation

A Control Flow is an orchestration of pipeline activities that includes chaining activities in a sequence, branching, defining parameters at the pipeline level, and passing arguments while invoking the pipeline on demand or from a trigger.

It also includes custom-state passing and looping containers, that are, For-each iterators. If a `ForEach` loop is used as a control flow activity, Azure Data Factory can start these multiple copy activities in parallel.

This allows you to build complex and iterative processing logic within the pipelines you create with Azure Data Factory. It supports diverse integration flows and patterns in the modern data warehouse, by enabling this flexible data pipeline model.

Chaining activities

Within Azure Data Factory you can chain activities in a sequence within a pipeline. It is possible to use the `dependsOn` property in an activity definition to chain it with an upstream activity.

Branching activities

Use Azure Data Factory for Branching activities within a pipeline. An example of a branching activity is *TheIf-condition* activity which is similar to an if-statement provided in programming languages. A branching activity evaluates a set of activities when the condition evaluates to true and another set of activities when the condition evaluates to false.

Parameters

You can define parameters at the pipeline level and pass arguments while you're invoking the pipeline on-demand or from a trigger. Activities can consume the arguments that are passed to the pipeline.

Custom state passing

Custom state passing is made possible with Azure Data Factory. Custom state passing is an activity that created output or the state of the activity that needs to be consumed by a subsequent activity in the pipeline. An example, is that in a JSON definition of an activity, you can access the output of the previous activity. Using custom state passing, enables you to build workflows where values are passing through activities.

Looping containers

The looping containers umbrella of control flow such as the `ForEach` activity defines repetition in a pipeline. It enables you to iterate over a collection and runs specified activities in the defined loop. It works similar as the 'for each looping structure' used in programming languages. Besides for each activity, there is also an Until activity. This functionality is similar to a do-until loop used in programming. What it does is running a set of activities (do) in a loop until the condition (until) is met.

Trigger based flows

Pipelines can be triggered by on-demand (event-based, i.e. blob post) or wall-clock time.

Invoke a pipeline from another pipeline

The Execute Pipeline activity with Azure Data Factory allows a Data Factory pipeline to invoke another pipeline.

Delta flows

Use-cases related to using delta flows, is delta loads. Delta loads in ETL patterns will only load data that has changed since a previous iteration of a pipeline. Capabilities such as lookup activity, and flexible scheduling helps handling delta load jobs. In case of using a Lookup activity, it will read or look up a record or table name value from any external source. This output can further be referenced by succeeding activities.

Other control flows

There are many more control flow activities. Below you find a couple of other useful activities.

• Web activity: The web activity in Azure Data Factory using control flows, can call a custom `RESTendpoint` from a Data Factory pipeline. Datasets and linked services can be passed in order to get consumed by the activity.

• Get metadata activity: The Get metadata activity retrieves the metadata of any data in Azure Data Factory.

https://docs.microsoft.com/en-us/azure/data-factory/introduction

Question 65: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

[?] for Storage provides an extra layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts. This layer of

protection allows you to address threats without being a security expert or managing security monitoring systems.

Security alerts are triggered when anomalies in activity occur. These security alerts are integrated with Azure Security Centre, and are also sent via email to subscription administrators, with details of suspicious activity and recommendations on how to investigate and remediate threats.

- Azure Defender
  (Correct)
- Azure RBAC
- Azure Shield
- Azure Armour
- Azure Vault

Explanation
Azure Defender for Storage provides an extra layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts. This layer of protection allows you to address threats without being a security expert or managing security monitoring systems.

Security alerts are triggered when anomalies in activity occur. These security alerts are integrated with Azure Security Centre, and are also sent via email to subscription administrators, with details of suspicious activity and recommendations on how to investigate and remediate threats.

Azure Defender for Storage is currently available for Blob storage, Azure Files, and Azure Data Lake Storage Gen2. Account types that support Azure Defender include general-purpose v2, block blob, and Blob storage accounts. Azure Defender for Storage is available in all public clouds and US government clouds, but not in other sovereign or Azure Government cloud regions.

Accounts with hierarchical namespaces enabled for Data Lake Storage support transactions using both the Azure Blob storage APIs and the Data Lake Storage APIs. Azure file shares support transactions over SMB.

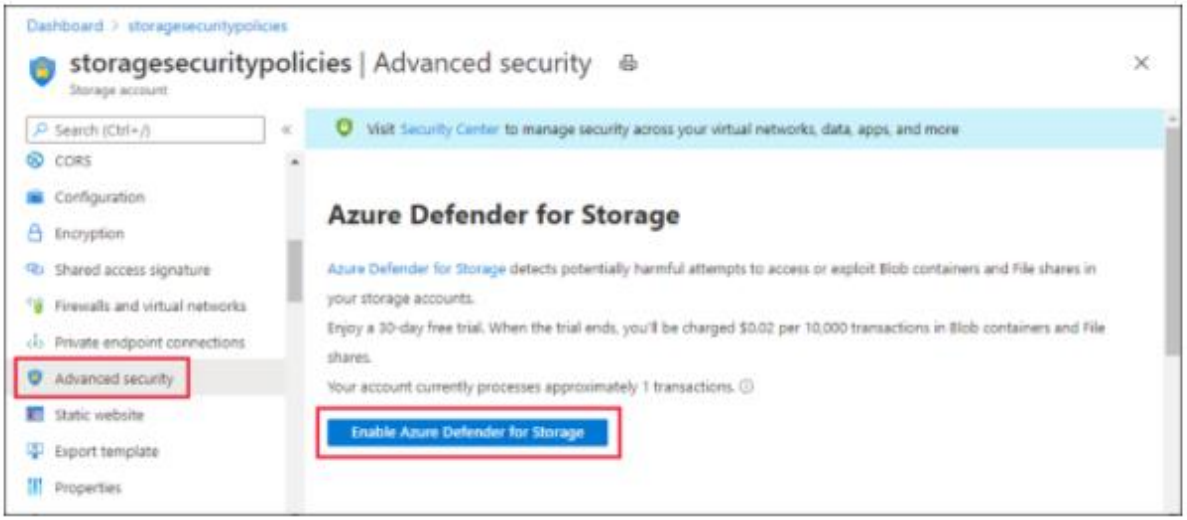
You can turn on Azure Defender for Storage in the Azure portal through the configuration page of the Azure Storage account, or in the advanced security section of the Azure portal.

Follow these steps.

1. Launch the Azure portal.

2. Navigate to your storage account. Under **Settings**, select **Advanced security**.

3. Select **Enable Azure Defender for Storage**.



https://docs.microsoft.com/en-us/azure/security-center/defender-for-storage-introduction

Question 66: Skipped

Scenario: You are working as a consultant at **Advanced Idea Mechanics** (**A.I.M.**) who is a privately funded think tank organized of a group of brilliant scientists whose sole dedication is to acquire and develop power through technological means. Their goal is to use this power to overthrow the governments of the world. They supply arms and technology to radicals and subversive organizations in order to foster a violent technological revolution of society while making a profit.

The company has 10,000 employees. Most employees are located in Europe. The company supports teams worldwide.

AIM has two main locations: a main office in London, England, and a manufacturing plant in Berlin, Germany.

During events, 100 engineers set up a remote portable office by using a VPN to connect the datacentre in the London office. The portable office is set up and torn down in approximately 20 different countries each year.

AIM runs Microsoft SQL Server in an on-premises virtual machine (VM).

Required:

• Migration of the database to Azure SQL Database

• Synchronize users from Active Directory to Azure Active Directory (Azure AD)

• Configure Azure SQL Database to use an Azure AD user as administrator

Which of the following should be configured?

- For each Azure SQL Database, set the Access Control to administrator.
- For each Azure SQL Database server, set the Access Control to administrator.
- For each Azure SQL Database, set the Active Directory administrator role. **(Correct)**
- For each Azure SQL Database server, set the Active Directory to administrator.

Explanation
There are two administrative accounts (Server admin and Active Directory admin) that act as administrators.

One Azure Active Directory account, either an individual or security group account, can also be configured as an administrator. It is optional to configure an Azure

AD administrator, but an Azure AD administrator must be configured if you want to use Azure AD accounts to connect to SQL Database.

Authentication and authorization

Authentication is the process of proving the user is who they claim to be. A user connects to a database using a user account. When a user attempts to connect to a database, they provide a user account and authentication information. The user is authenticated using one of the following two authentication methods:

• SQL authentication.

With this authentication method, the user submits a user account name and associated password to establish a connection. This password is stored in the master database for user accounts linked to a login or stored in the database containing the user accounts *not* linked to a login.

• Azure Active Directory Authentication

With this authentication method, the user submits a user account name and requests that the service use the credential information stored in Azure Active Directory (Azure AD).

Logins and users: A user account in a database can be associated with a login that is stored in the master database or can be a user name that is stored in an individual database.

• A **login** is an individual account in the master database, to which a user account in one or more databases can be linked. With a login, the credential information for the user account is stored with the login.

• A **user account** is an individual account in any database that may be, but does not have to be, linked to a login. With a user account that is not linked to a login, the credential information is stored with the user account.

Authorization to access data and perform various actions are managed using database roles and explicit permissions. Authorization refers to the permissions assigned to a user, and determines what that user is allowed to do. Authorization is controlled by your user account's database role memberships and object-level permissions. As a best practice, you should grant users the least privileges necessary.

Existing logins and user accounts after creating a new database

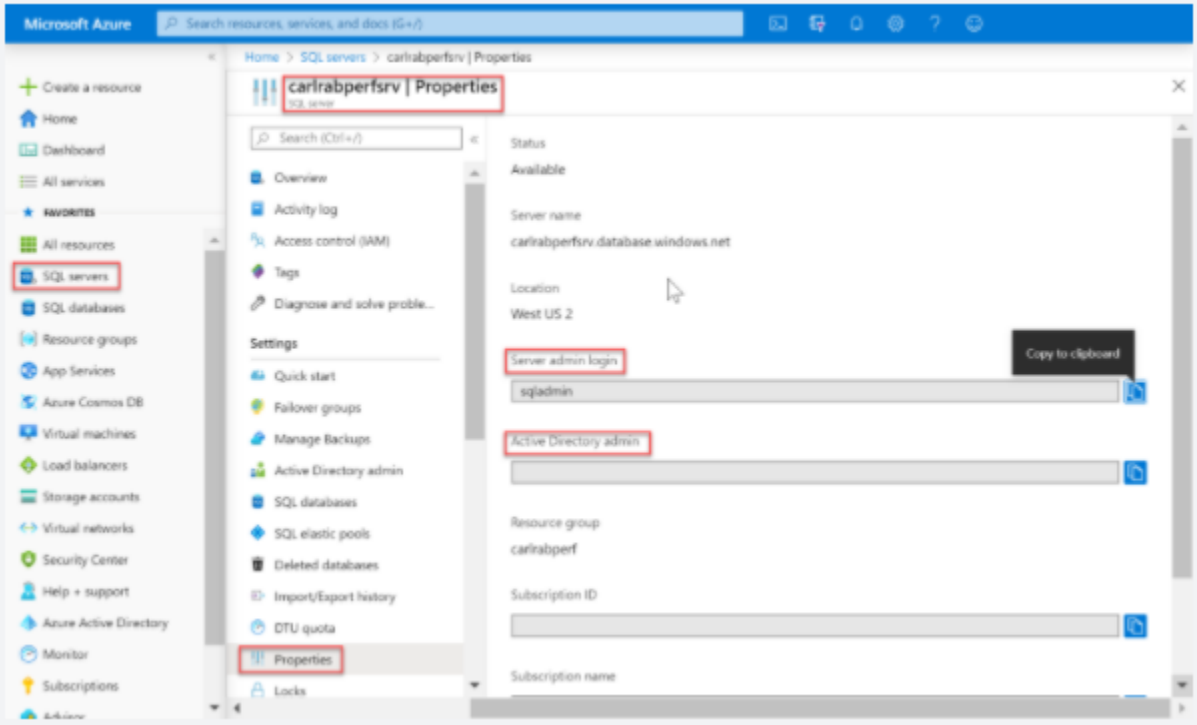
When you first deploy Azure SQL, you specify an admin login and an associated password for that login. This administrative account is called **Server admin**. The following configuration of logins and users in the master and user databases occurs during deployment:

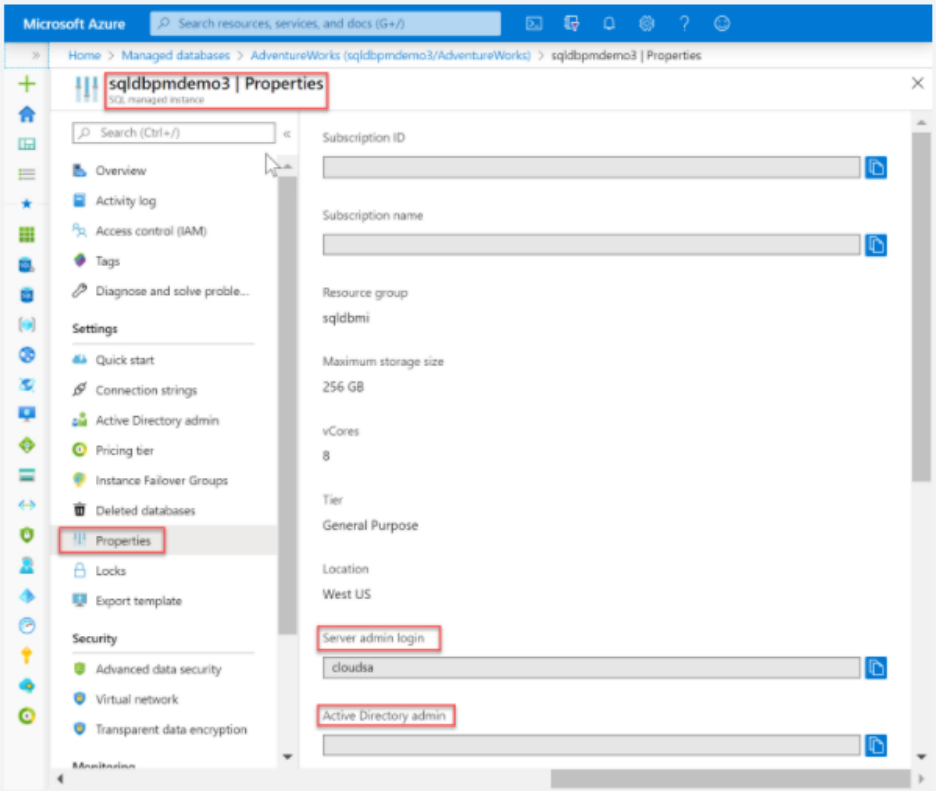
• A SQL login with administrative privileges is created using the login name you specified. A login is an individual user account for logging in to SQL Database, SQL Managed Instance, and Azure Synapse.

• This login is granted full administrative permissions on all databases as a server-level principal. The login has all available permissions and can't be limited. In a SQL Managed Instance, this login is added to the sysadmin fixed server role (this role does not exist in Azure SQL Database).

• A user account called `dbo` is created for this login in each user database.
The dbo user has all database permissions in the database and is mapped to the `db_owner` fixed database role. Additional fixed database roles are discussed later in this article.

To identify the administrator accounts for a database, open the Azure portal, and navigate to the **Properties** tab of your server or managed instance.





*Important: The admin login name can't be changed after it has been created. To reset the password for the server admin, go to the Azure portal, click **SQL Servers**, select the server from the list, and then click **Reset Password**. To reset the password for the SQL Managed Instance, go to the Azure portal, click the instance, and click **Reset password**. You can also use PowerShell or the Azure CLI.*

Create additional logins and users having administrative permissions

At this point, your server or managed instance is only configured for access using a single SQL login and user account. To create additional logins with full or partial administrative permissions, you have the following options (depending on your deployment mode):

Create an Azure Active Directory administrator account with full administrative permissions

Enable Azure Active Directory authentication and create an Azure AD administrator login. One Azure Active Directory account can be configured as an administrator of the Azure SQL deployment with full administrative permissions. This account can be either an individual or security group account. An Azure AD administrator **must** be configured if you want to use Azure AD accounts to connect to SQL Database, SQL Managed Instance, or Azure Synapse. For detailed information on enabling Azure AD authentication for all Azure SQL deployment types, see the following articles:

• Use Azure Active Directory authentication for authentication with SQL

• Configure and manage Azure Active Directory authentication with SQL

In SQL Managed Instance, create SQL logins with full administrative permissions

• Create an additional SQL login in the master database.

• Add the login to the sysadmin fixed server role using the ALTER SERVER ROLE statement. This login will have full administrative permissions.

• Alternatively, create an Azure AD login using the CREATE LOGIN syntax.

In SQL Database, create SQL logins with limited administrative permissions

• Create an additional SQL login in the master database.

• Create a user account in the master database associated with this new login.

• Add the user account to the `dbmanager`, the `loginmanager` role, or both in the `master` database using the ALTER ROLE statement (for Azure Synapse, use the sp_addrolemember statement).

Members of these special master database roles for Azure SQL Database have authority to create and manage databases or to create and manage logins. In databases created by a user that is a member of the `dbmanager` role, the member is mapped to the `db_owner` fixed database role and can log into and manage that database using the `dbo` user account. These roles have no explicit permissions outside of the master database.

https://docs.microsoft.com/en-us/azure/sql-database/sql-database-manage-logins

Question 67: Skipped

Scenario: Your team has deployed a factory to production and realizes there's a bug that needs to be fixed right away, but you can't deploy the current collaboration branch.

What is the best action to take?

- None of the listed options
- Create a rollback to a savepoint
- Deploy a hotfix
  (Correct)
- Deploy a timeshift
- Utilize a workhole

Explanation
Hotfix production environment

If you deploy a factory to production and realize there's a bug that needs to be fixed right away, but you can't deploy the current collaboration branch, you might need to deploy a hotfix. This approach is as known as quick-fix engineering or QFE.

1. In Azure DevOps, go to the release that was deployed to production. Find the last commit that was deployed.

2. From the commit message, get the commit ID of the collaboration branch.

3. Create a new hotfix branch from that commit.

4. Go to the Azure Data Factory UX and switch to the hotfix branch.

5. By using the Azure Data Factory UX, fix the bug. Test your changes.

6. After the fix is verified, select **Export ARM Template** to get the hotfix Resource Manager template.

7. Manually check this build into the publish branch.

8. If you've configured your release pipeline to automatically trigger based on adf_publish check-ins, a new release will start automatically. Otherwise, manually queue a release.

9. Deploy the hotfix release to the test and production factories. This release contains the previous production payload plus the fix that you made in step 5.

10. Add the changes from the hotfix to the development branch so that later releases won't include the same bug.

https://docs.microsoft.com/en-us/azure/data-factory/continuous-integration-deployment

Question 68: Skipped

Scenario: You are working at an online retailer and have been tasked with finding average of sales transactions by storefront.

Which of the following aggregates would you use?

- `df.select(col("storefront")).avg("completedTransactions")`
- `df.select(col("storefront")).avg("completedTransactions").groupBy(col("storefront"))`
- `df.groupBy(col("storefront")).avg("completedTransactions")`
  (Correct)
- `df.groupBy(col("storefront")).avg(col("completedTransactions"))`

Explanation

The syntax `df.groupBy(col("storefront")).avg("completedTransactions")` groups the data by the storefront Column, then calculates the average value of completed sales transactions.

https://docs.databricks.com/spark/latest/dataframes-datasets/introduction-to-dataframes-python.html

Question 69: Skipped

Which is the correct syntax for overwriting data in Azure Synapse Analytics from a Databricks notebook?

- `df.write.format("com.databricks.spark.sqldw").mode("overwrite").option("...").option("...").save()`
  (Correct)
- `df.write.mode("overwrite").option("...").option("...").save()`
- `df.write.format("com.databricks.spark.sqldw").update().option("...").option("...").save()`
- `df.write.format("com.databricks.spark.sqldw").overwrite().option("...").option("...").save()`

Explanation

`df.write.format("com.databricks.spark.sqldw").mode("overwrite").option("...").option("...").save()` is the correct syntax for overwriting data in Azure Synapse Analytics from a Databricks notebook.

The key is to specify the correct format, intended write mode, and options that specify the Azure Synapse Analytics properties.

https://docs.microsoft.com/en-us/azure/databricks/delta/delta-batch

Question 70: Skipped

The following are the facets of Azure Databricks security:

• Data Protection

• IAM/Auth

• Network

• Compliance

Which of the following comprise Data Protection within Azure Databricks security? (Select five)

- ☐ VNet Injection
- ☐ Managed Keys **(Correct)**
- ☐ TLS **(Correct)**
- ☐ Azure VNet service endpoints
- ☐ ACLs **(Correct)**
- ☐ AAD **(Correct)**
- ☐ Vault Secrets **(Correct)**
- ☐ Azure Private Link
- ☐ VNet Peering

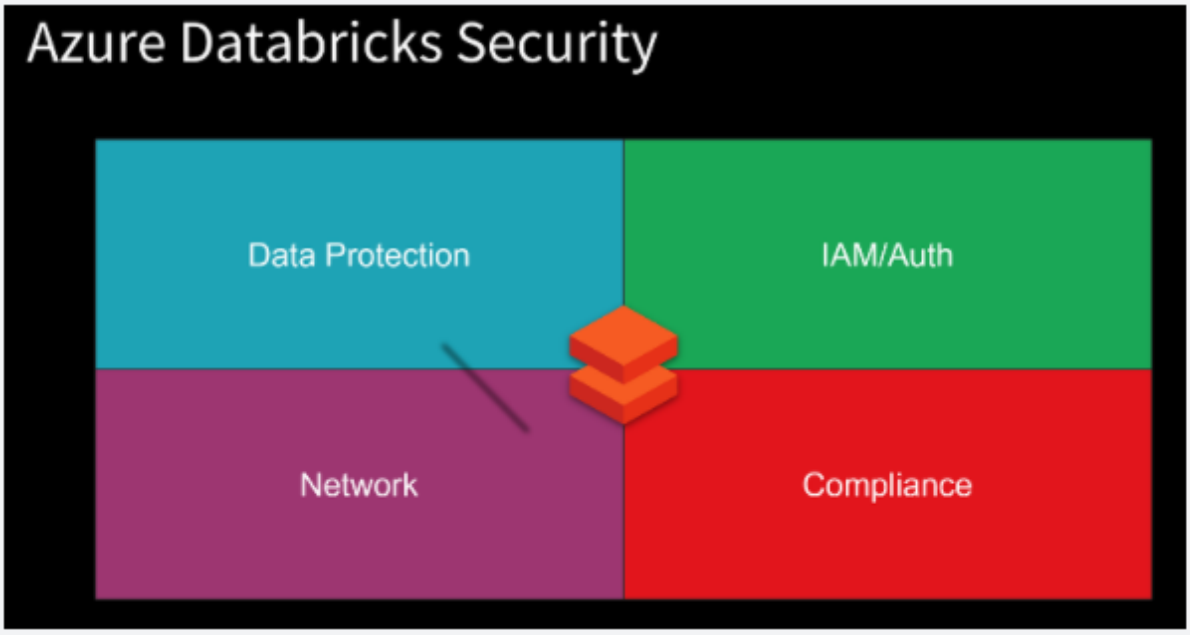
Explanation
The following are the facets of Azure Databricks security:

• Data Protection

• IAM/Auth

• Network

• Compliance



Data Protection is comprised of the following:

• Encryption at-rest – Service Managed Keys, User Managed Keys

• Encryption in-transit (Transport Layer Security - TLS)

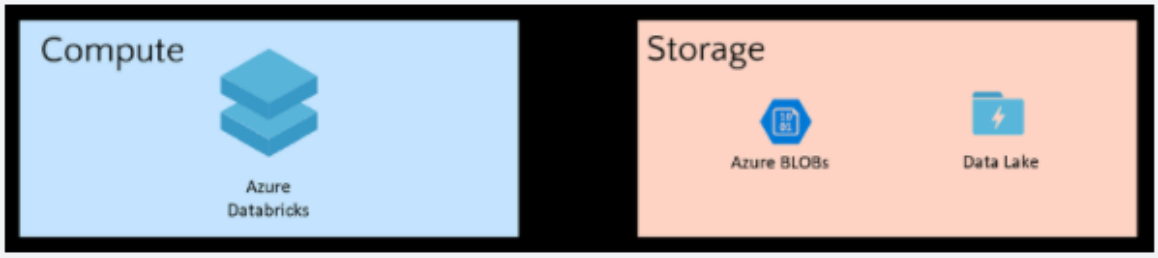
• File/Folder Level access control lists (ACLs) for Azure Active Directory (AAD) Users, Groups, Service Principals

• ACLs for Clusters, Folders, Notebooks, Tables, Jobs

• Secrets with Azure Key Vault

Encryption at-rest

Azure Databricks has separation of compute and storage.



Azure Databricks is a compute platform. It does not store data, except for notebooks. Clusters are transient in nature. They process the data then are terminated. All data is stored in the customer's subscription. Because the Azure storage services use server-side encryption, communication between these services and the Databricks clusters is seamless.

Storage Services such as Azure Storage Blobs and Azure Data Lake Storage (Gen1/2) provide:

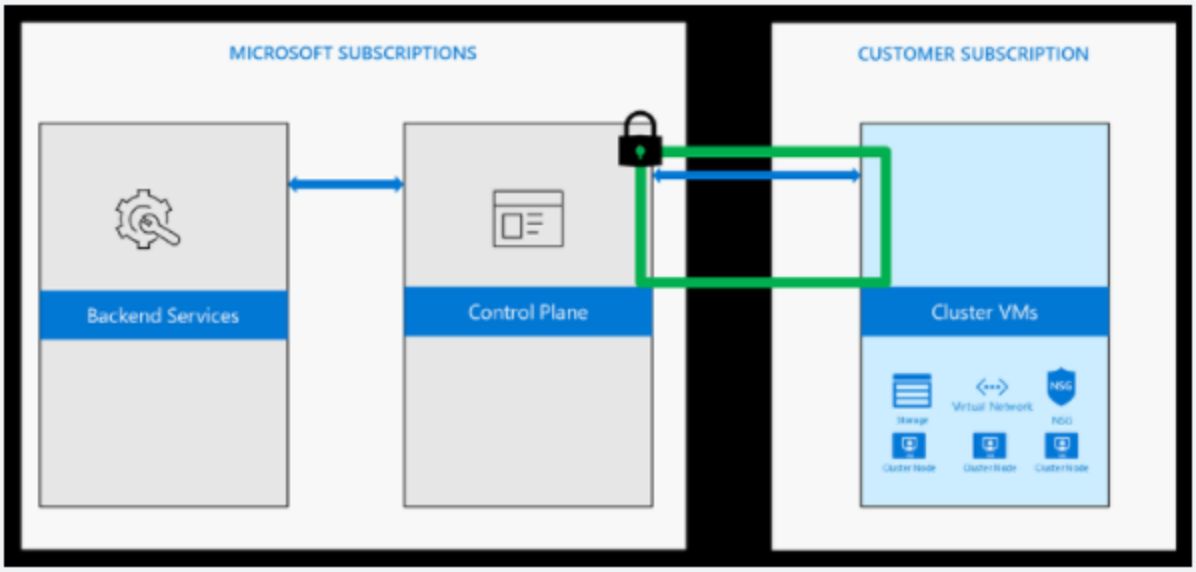
• Encryption of Data - Automatic server-side encryption in addition to encryption on storage attached to the VMs

• Customer Managed Keys - Bring your own keys with Key Vault integration

• File/Folder Level ACLs (Azure Data Lake Storage (Gen1/2))

Encryption in-transit

All the traffic from the Control Plane to the clusters in the customer subscription (Data Plane) is always encrypted with TLS.



When clusters access data from various Azure services, TLS is always used to ensure encryption in-transit.

When customers access notebooks via their web browsers, the connection is also secured with TLS.

Access control - ADLS Passthrough

When enabled, authentication automatically takes place in Azure Data Lake Storage (ADLS) from Azure Databricks clusters using the same Azure Active Directory (Azure AD) identity that one uses to log into Azure Databricks. Commands running on a configured cluster will be able to read and write data in ADLS without needing to configure service principal credentials. Any ACLs applied at the folder or file level in ADLS are enforced based on the user's identity.

ADLS Passthrough is configured when you create a cluster in the Azure Databricks workspace. ADLS Gen1 requires Databricks Runtime 5.1+. ADLS Gen2 requires 5.3+.

On a *standard cluster*, when you enable this setting you must set single user access to one of the Azure Active Directory (AAD) users in the Azure Databricks workspace. Only one user is allowed to run commands on this cluster when Credential Passthrough is enabled.

High-concurrency clusters can be shared by multiple users. When you enable ADLS Passthrough on this type of cluster, it does not require you to select a single user.

Access control - Folders

Access control is available only in the Premium SKU. By default, all users can create and modify workspace objects unless an administrator enables workspace access control. With workspace access control, individual permissions determine a user's abilities. This section describes the individual permissions and how to enable and configure workspace access control.

You can assign five permission levels to notebooks and folders: No Permissions, Read, Run, Edit, and Manage. The following tables lists the abilities for each permission.

Ability	No Permissions	Read	Run	Edit	Manage
View items		X	X	X	X
Create, clone, import, export items		X	X	X	X
Run commands on notebooks			X	X	X
Attach/detach notebooks			X	X	X
Delete items				X	X
Move/rename items				X	X
Change permissions					X

Access control - Notebooks

Ability	No Permissions	Read	Run	Edit	Manage
View cells		X	X	X	X
Comment		X	X	X	X
Run commands			X	X	X
Attach/detach notebooks			X	X	X
Edit cells				X	X
Change permissions					X

All notebooks in a folder inherit all permissions settings of that folder. For example, a user that has Run permission on a folder has Run permission on all notebooks in that folder.

To enable workspace access control:

• Go to the Admin Console.

• Select the Access Control tab.

• Click the Enable button next to Workspace Access Control.

• Click Confirm to confirm the change.

Access control - Clusters

All users can view libraries. To control who can attach libraries to clusters, manage access control on clusters.

By default, all users can create and modify clusters unless an administrator enables cluster access control. With cluster access control, permissions determine a user's abilities. There are four permission levels for a cluster: No Permissions, Can Attach To, Can Restart, and Can Manage:

Ability	No Permissions	Can Attach To	Can Restart	Can Manage
Attach notebook to cluster		x	x	x
View Spark UI		x	x	x
View cluster metrics		x	x	x
Terminate cluster			x	x
Start cluster			x	x
Restart cluster			x	x
Edit cluster				x
Attach library to cluster				x
Resize cluster				x
Modify permissions				x

Note: You have Can Manage permission for any cluster that you create.

Access control - Jobs

To control who can run jobs and see the results of job runs, manage access control on jobs.

There are five permission levels for jobs: No Permissions, Can View, Can Manage Run, Is Owner, and Can Manage. The Can Manage permission is reserved for administrators.

Ability	No Permissions	Can View	Can Manage Run	Is Owner	Can Manage (admin)
View job details and settings	x	x	x	x	x
View results, Spark UI, logs of a job run		x	x	x	x
Run now			x	x	x
Cancel run			x	x	x
Edit job settings				x	x
Modify permissions				x	x

Access control - Tables

Table access control (table ACLs) lets you programmatically grant and revoke access to your data from SQL, Python, and PySpark.

By default, all users have access to all data stored in a cluster's managed tables unless an administrator enables table access control for that cluster. Once table access control is enabled for a cluster, users can set permissions for data objects on that cluster.

Before you can grant or revoke privileges on data objects, an administrator must enable table access control for the cluster.

View-based access control model

The Azure Databricks view-based access control model defines the following privileges:

- `SELECT` – gives read access to an object.
- `CREATE` – gives ability to create an object (for example, a table in a database)
- `MODIFY` – gives ability to add/delete/modify data to/from an object.
- `READ_METADATA` – gives ability to view an object and its metadata.
- `CREATE_NAMED_FUNCTION` – gives ability to create a named UDF in an existing catalogue or database.
- `ALL PRIVILEGES` – gives all privileges (gets translated into all the above privileges)

The privileges above can apply to the following classes of objects:

- `CATALOG` - controls access to the entire data catalog.
- `DATABASE` - controls access to a database.
- `TABLE` - controls access to a managed or external table.
- `VIEW` - controls access to SQL views.
- `FUNCTION` - controls access to a named function.
- `ANONYMOUS FUNCTION` - controls access to anonymous or temporary functions.
- `ANY FILE` - controls access to the underlying filesystem.

Secrets

Using the Secrets APIs, Secrets can be securely stored including in an Azure Key Vault or Databricks backend. Authorized users can consume the secrets to access services.

Azure Databricks has two types of secret scopes: Key Vault-backed and Databricks-backed. These secret scopes allow you to store secrets, such as database connection strings, securely. If someone tries to output a secret to a notebook, it is replaced by `[REDACTED]`. This helps prevent someone from viewing the secret or accidentally leaking it when displaying or sharing the notebook.

As a best practice, instead of directly entering your credentials into a notebook, use Azure Databricks secrets to store your credentials and reference them in notebooks and jobs.

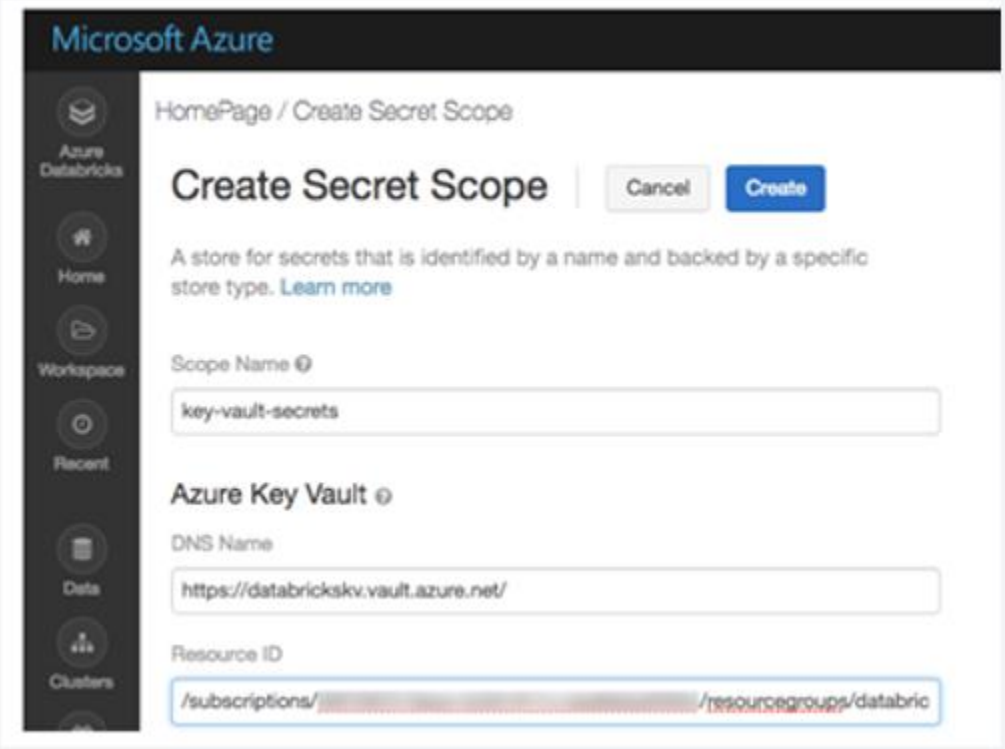
To set up secrets you:

• Create a secret scope. Secret scope names are case insensitive.

• Add secrets to the scope. Secret names are case insensitive.

• If you have the Azure Databricks Premium Plan, assign access control to the secret scope.

Screenshot of creating an Azure Key Vault-backed secret scope:



https://docs.microsoft.com/en-us/azure/databricks/scenarios/security-baseline

Question 71: Skipped

Scenario: You are working on a project using Azure Synapse Studio and want to configure a private endpoint. You open up Azure Synapse Studio, go to the manage hub, and see that the private endpoints is greyed out.

Why is the option not available?

- A managed virtual network has not been created.
  (Correct)
- Azure Synapse Studio does not support the creation of private endpoints.
- There are service interruptions which must be troubleshot first.
- A conditional access policy has to be defined first.

Explanation
In order to create a private endpoint, you first must create a managed virtual network.

https://docs.microsoft.com/en-us/azure/synapse-analytics/security/synapse-private-link-hubs

Question 72: Skipped

Scenario: You have been contracted by Wayne Enterprises, a company owned by Bruce Wayne with market value of over twenty seven million dollars. Bruce founded Wayne Enterprises shortly after he created the Wayne Foundation and he became the president and chairman of the company.

Bruce has come to you because his IT team needs advice on the configuration and synchronization of data between an on-premises Microsoft SQL Server database to Azure SQL Database.

Recently, ad-hoc and reporting queries are being overutilized on the on-premises production instance and your expert advise is required on the following points.

Requirements:

• Execute an initial data synchronization to Azure SQL Database (minimize downtime)

• Execute bi-directional data synchronization after initial synchronization

A synchronization solution must be created and implemented and Bruce and the team look to you as the Azure expert. Which synchronization method should you advise the team to use?

- Azure SQL Data Sync
  (Correct)
- Backup and restore
- SQL Server Agent job
- Data Migration Assistant
- Transactional replication

Explanation
SQL Data Sync is a service built on Azure SQL Database that lets you synchronize the data you select bi-directionally across multiple databases, both on-premises and in the cloud.

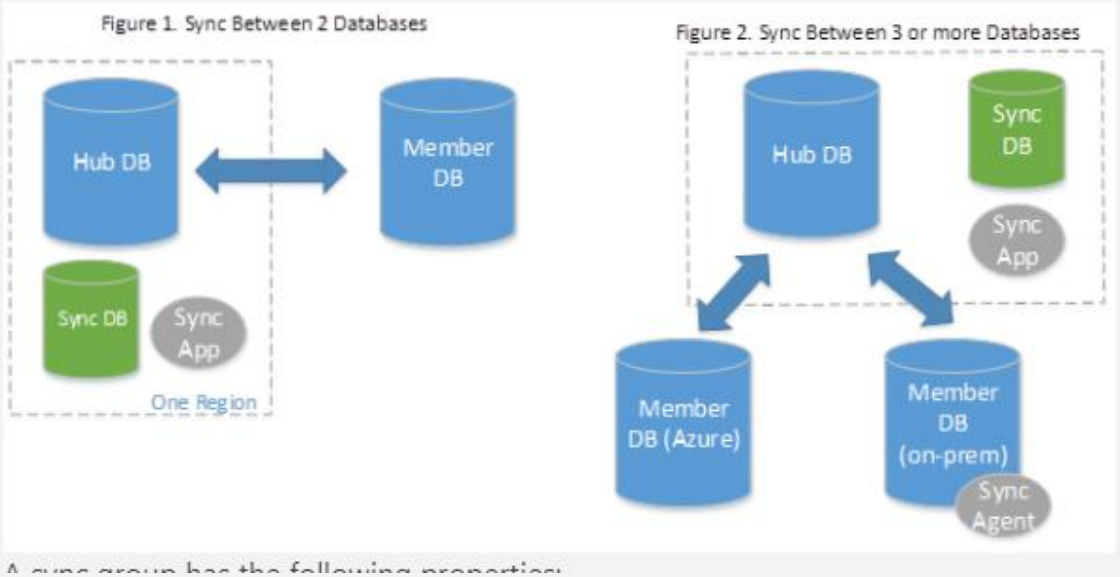
Data Sync is based around the concept of a sync group. A sync group is a group of databases that you want to synchronize.

Data Sync uses a hub and spoke topology to synchronize data. You define one of the databases in the sync group as the hub database. The rest of the databases are member databases. Sync occurs only between the hub and individual members.

• The **Hub Database** must be an Azure SQL Database.

• The **member databases** can be either databases in Azure SQL Database or in instances of SQL Server.

• The **Sync Metadata Database** contains the metadata and log for Data Sync. The Sync Metadata Database has to be an Azure SQL Database located in the same region as the Hub Database. The Sync Metadata Database is customer created and customer owned. You can only have one Sync Metadata Database per region and subscription. Sync Metadata Database cannot be deleted or renamed while sync groups or sync agents exist. Microsoft recommends to create a new, empty database for use as the Sync Metadata Database. Data Sync creates tables in this database and runs a frequent workload.



A sync group has the following properties:

• The **Sync Schema** describes which data is being synchronized.

• The **Sync Direction** can be bi-directional or can flow in only one direction. That is, the Sync Direction can be *Hub to Member*, or *Member to Hub*, or both.

• The **Sync Interval** describes how often synchronization occurs.

• The **Conflict Resolution Policy** is a group level policy, which can be *Hub wins* or *Member wins*.

Server instances.

With Data Sync, you can keep data synchronized between your on-premises databases and Azure SQL databases to enable hybrid applications. Here are the main use cases for Data Sync:

• **Hybrid Data Synchronization:** With Data Sync, you can keep data synchronized between your databases in SQL Server and Azure SQL Database to enable hybrid applications. This capability may appeal to customers who are considering moving to the cloud and would like to put some of their application in Azure.

• **Distributed Applications:** In many cases, it's beneficial to separate different workloads across different databases. For example, if you have a large production database, but you also need to run a reporting or analytics workload on this data, it's helpful to have a

second database for this additional workload. This approach minimizes the performance impact on your production workload. You can use Data Sync to keep these two databases synchronized.

• **Globally Distributed Applications:** Many businesses span several regions and even several countries/regions. To minimize network latency, it's best to have your data in a region close to you. With Data Sync, you can easily keep databases in regions around the world synchronized.

Compare Data Sync with Transactional Replication

	Data Sync	Transactional Replication
Advantages	- Active-active support - Bi-directional between on-premises and Azure SQL Database	- Lower latency - Transactional consistency - Reuse existing topology after migration -Azure SQL Managed Instance support
Disadvantages	- No transactional consistency - Higher performance impact	- Can't publish from Azure SQL Database - High maintenance cost

https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-data-sync-data-sql-server-sql-database

Question 73: Skipped

Connectors are Azure Data Factory objects that enable your Linked Services and Datasets to connect to a wide variety of data sources and sinks. These can include connections to Azure resources and third-party connectors such as Amazon S3 or Google cloud. There are nearly 100 connectors that are available.

Which are among the file formats supported? (Select six)

- ☐ XLSB format
- ☐ JSON format **(Correct)**
- ☐ TXT format
- ☐

Delimited text format
(Correct)

- CSV format
- Avro format
(Correct)
- Binary format
(Correct)
- Parquet format
(Correct)
- ORC format
(Correct)
- XLSX format
- PDF format
- XLS format

Explanation
Connectors are Azure Data Factory objects that enable your Linked Services and Datasets to connect to a wide variety of data sources and sinks. These can include connections to Azure resources and third-party connectors such as Amazon S3 or Google cloud. There are nearly 100 connectors that are available, and they work with the Copy, Data Flow, Look up, Get Metadata, and Delete activities that can be found within Azure Data Factory.

The file formats that are supported include:

• Avro format

• Binary format

• Delimited text format

• JSON format

• ORC format

• Parquet format

There are too many data stores to list, but the following lists the categories of data stores and two examples of the types of connectors that exist.

Category: Azure

Data Store example: Azure Data Lake Store, Azure Synapse Analytics

Category: Databases

Data Store example: Netezza, Greenplum

Category: NoSQL stores

Data Store example: Cassandra, MongoDB

Category: File

Data Store example: FTP, Google Cloud Storage

Category: Generic protocols

Data Store example: REST, ODBC

Category: Services & Apps

Data Store example: Dynamics, SalesForce

The list of connectors is constantly evolving. You can keep up to date with the latest list, and the activity support by looking at the connectors overview page

https://docs.microsoft.com/en-us/azure/data-factory/connector-overview

Question 74: Skipped

Azure Synapse Analytics has a rich set of tools and methods available to load data into SQL Pools.

Which of the following are you able to load into Azure Synapse Analytics? (Select all that apply)

- ☐ Data streams **(Correct)**
- ☐ On-premises **(Correct)**
- ☐ Semi-structured data **(Correct)**
- ☐ Non-relational datastores **(Correct)**
- ☐ Structured data **(Correct)**
- ☐ Non-Azure clouds **(Correct)**
- ☐ Data batches **(Correct)**
- ☐ Relational datastores **(Correct)**

Explanation
Azure Synapse Analytics has a rich set of tools and methods available to load data into SQL Pools. You can load data from relational or non-relational datastores; structured or semi-structured; on premises systems or other clouds; in batches or streams.

Based on the variety of data that you work with, your data loads can include:

Data loads directly from Azure storage with transact-sql and the copy statement

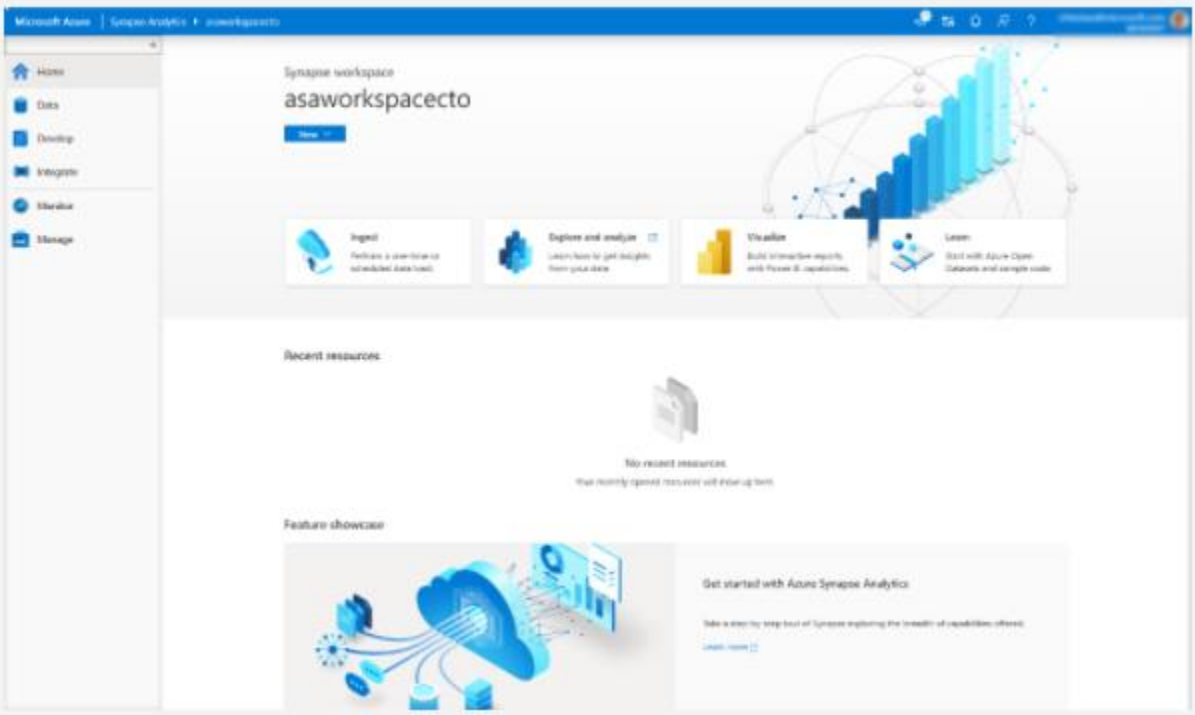
Within Azure Synapse Studio, you can write Transact-SQL code that runs against any configured SQL Pools within the workspace. Similarly, within the same Transact-SQL script, you can read and digest data from Azure Blob Storage or Azure Data Lake and insert it into a table within the SQL Pool.

Perform data loads using Azure synapse pipeline data flows.

Data flows are a key feature within the Azure Synapse Studio experience. You can access the data flows from the Integrate hub. From within the Develop hub, you're able to access configured source repositories and run transformations against them to a variety of destinations referred to as sinks.

Use polybase by defining external tables

Using Transact-SQL, you can use PolyBase to access files that are located directly on Azure Storage as if they were structured tables within your SQL Pool. You define an **external data source** pointing to the location of the file or the folder the files reside in, the external file format, which can be GZip compressed delimited text, ORC, Parquet or JSON, and then the external table with the column attributes that map to the structure from the external files.



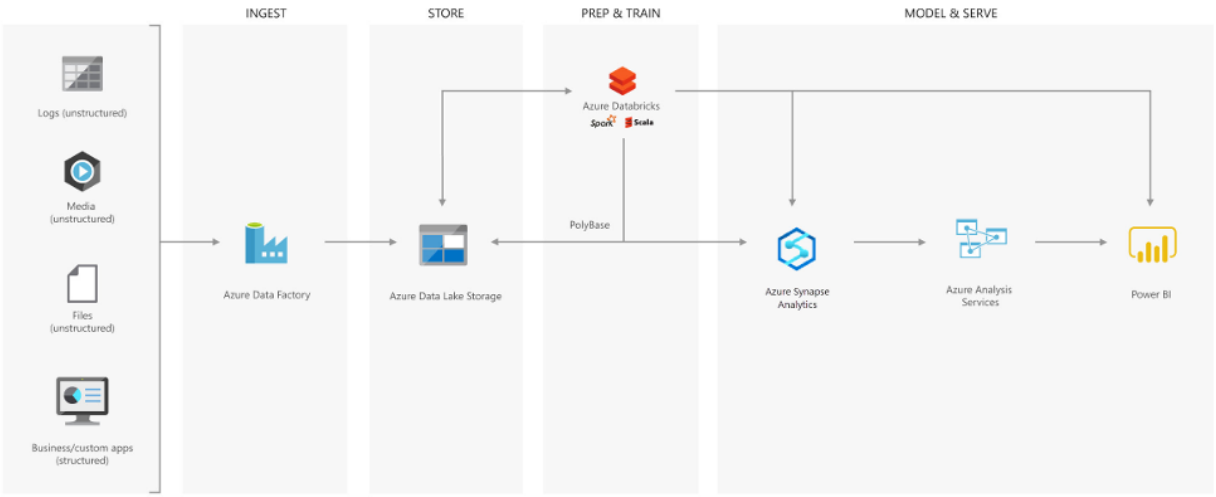
https://docs.microsoft.com/en-us/azure/data-factory/load-azure-sql-data-warehouse

Question 75: Skipped

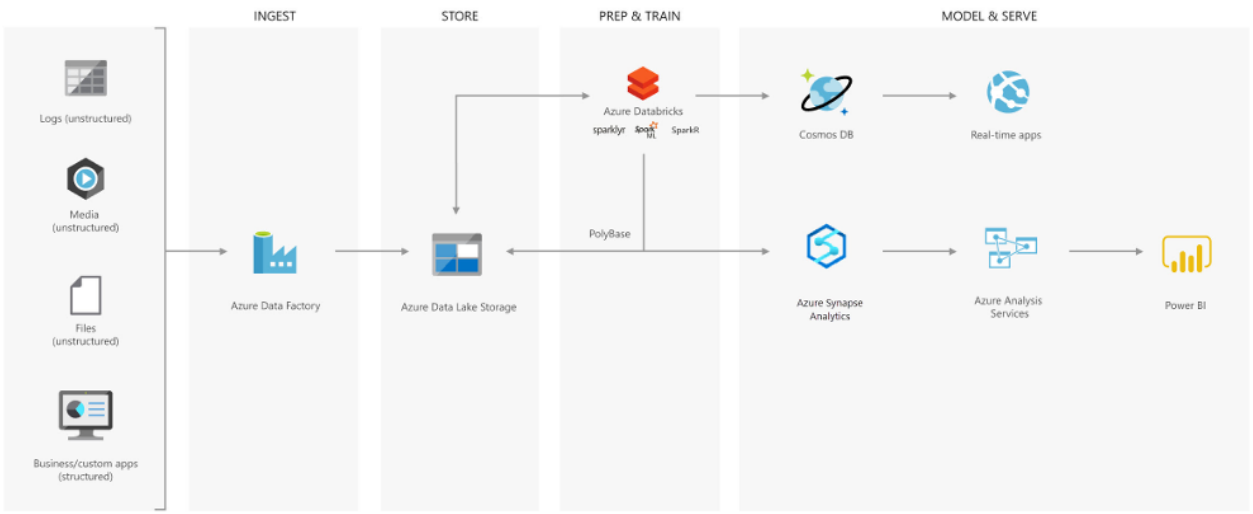
Scenario: You are a Data Engineer for HAMMER Industries, an organization that is working with a transport company to monitor the fleet of Heavy Goods Vehicles (HGV) that drive around Europe. Each HGV is equipped with sensor hardware that will continuously report metric data on the temperature, the speed, and the oil and brake solution levels of an HGV. When the engine is turned off, the sensor also outputs a file with summary information about a trip, including the mileage and elevation of a trip. A trip is a period in which the HGV engine is turned on and off.

Both the real-time data and batch data is processed in a machine learning model to predict a maintenance schedule for each of the HGVs. This data is made available to the downstream application that third-party garage companies can use if an HGV breaks down anywhere in Europe. In addition, historical reports about the HGV should be visually presented to users.

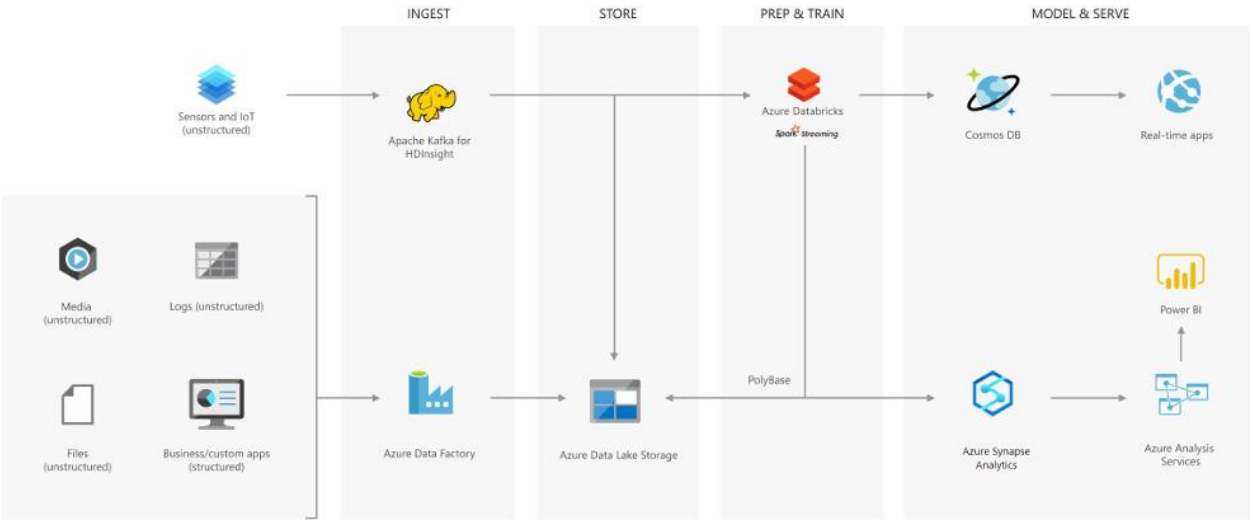
Design A:



Design B:



Design C:



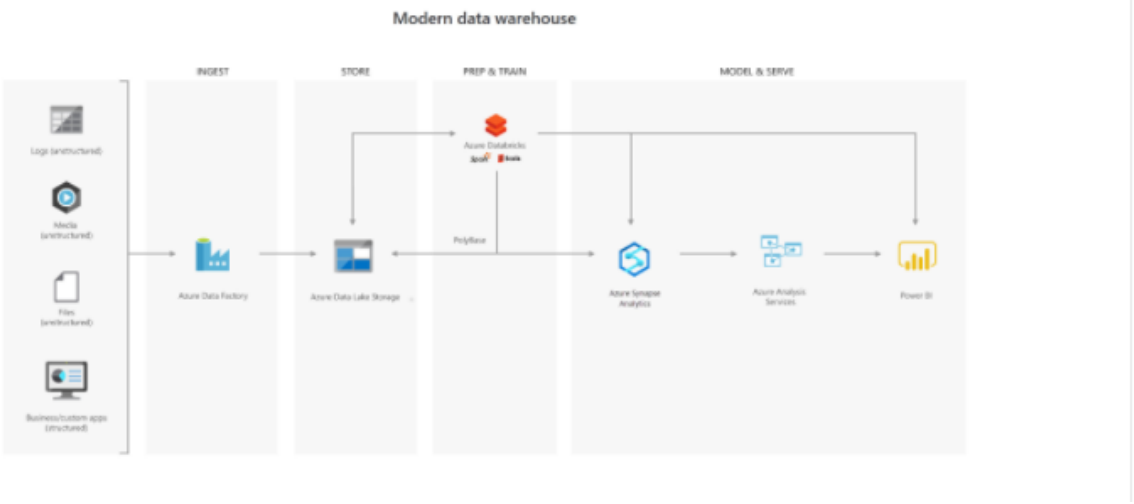
Which architecture would be best suited for the need?

- Design C
  (Correct)
- Design B
- None of the listed options
- Design A

Explanation
Creating a modern data warehouse

Imagine you're a Data Engineering consultant for a Avengers Security. In the past, they've created an on-premises business intelligence solution that used a Microsoft SQL Server Database Engine, SQL Server Integration Services, SQL Server Analysis Services, and SQL Server Reporting Services to provide historical reports. They tried using the Analysis Services Data Mining component to create a predictive analytics solution to predict the buying behaviour of customers. While this approach worked well with low volumes of data, it couldn't scale after more than a gigabyte of data was collected. Furthermore, they were never able to deal with the JSON data that a third-party application generated when a customer used the feedback module of the point of sale (POS) application.

The company has turned to you for help with creating an architecture that can scale with the data needs that are required to create a predictive model and to handle the JSON data so that it's integrated into the BI solution. You suggest the following architecture:



The architecture uses Azure Data Lake Storage at the centre of the solution for a modern data warehouse. Integration Services is replaced by Azure Data Factory to ingest data into the Data Lake from a business application. This is the source for the predictive model that is built into Azure Databricks. PolyBase is used to transfer the historical data into a big data relational format that is held in Azure Synapse Analytics, which also stores the results of the trained model from Databricks. Azure Analysis

Services provides the caching capability for SQL Data Warehouse to service many users and to present the data through Power BI reports.

Advanced analytics for big data

In this second use case, Azure Data Lake Storage plays an important role in providing a large-scale data store. Your skills are needed by Hydra Corporation, which is a global seller of bicycles and cycling components through a chain of resellers and on the internet. As their customers browse the product catalogue on their websites and add items to their baskets, a recommendation engine that is built into Azure Databricks recommends other products. They need to make sure that the results of their recommendation engine can scale globally. The recommendations are based on the web log files that are stored on the web servers and transferred to the Azure Databricks model hourly. The response time for the recommendation should be less than 1 ms. You propose the following architecture:

Real-time analytical solutions

To perform real-time analytical solutions, the ingestion phase of the architecture is changed for processing big data solutions. In this architecture, note the introduction of Apache Kafka for Azure HDInsight to ingest streaming data from an Internet of Things (IoT) device, although this could be replaced with Azure IoT Hub and Azure Stream Analytics. The key point is that the data is persisted in Data Lake Storage Gen2 to service other parts of the solution.

In this use case, you are a Data Engineer for HAMMER Industries, an organization that is working with a transport company to monitor the fleet of Heavy Goods Vehicles (HGV) that drive around Europe. Each HGV is equipped with sensor hardware that will continuously report metric data on the temperature, the speed, and the oil and brake solution levels of an HGV. When the engine is turned off, the sensor also outputs a file with summary information about a trip, including the mileage and elevation of a trip. A trip is a period in which the HGV engine is turned on and off.

Both the real-time data and batch data is processed in a machine learning model to predict a maintenance schedule for each of the HGVs. This data is made available to the downstream application that third-party garage companies can use if an HGV breaks down anywhere in Europe. In addition, historical reports about the HGV should be visually presented to users. As a result, the following architecture is proposed:

In this architecture, there are two ingestion streams. Azure Data Factory ingests the summary files that are generated when the HGV engine is turned off. Apache Kafka provides the real-time ingestion engine for the telemetry data. Both data streams are stored in Azure Data Lake Store for use in the future, but they are also passed on to other technologies to meet business needs. Both streaming and batch data are provided to the predictive model in Azure Databricks, and the results are published to Azure Cosmos DB to be used by the third-party garages. PolyBase transfers data from the Data Lake Store into SQL Data Warehouse where Azure Analysis Services creates the HGV reports by using Power BI.

https://docs.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-introduction

Question 76: Skipped

Which Azure Service is Azure Synapse Pipelines based on?

- None of the listed options
(Correct)

- Azure Data Explorer
- Azure Synapse Spark pools
- Azure Data Warehouse
- Azure Stream Analytics
- Azure Synapse Studio
- Azure Synapse Link

Explanation
Azure Synapse Pipelines is based in the Azure Data Factory service.

A data factory can have one or more pipelines. A pipeline is a logical grouping of activities that together perform a task. For example, a pipeline could contain a set of activities that ingest and clean log data, and then kick off a mapping data flow to analyze the log data. The pipeline allows you to manage the activities as a set instead of each one individually. You deploy and schedule the pipeline instead of the activities independently.

The activities in a pipeline define actions to perform on your data. For example, you may use a copy activity to copy data from SQL Server to an Azure Blob Storage. Then, use a data flow activity or a Databricks Notebook activity to process and transform data from the blob storage to an Azure Synapse Analytics pool on top of which business intelligence reporting solutions are built.

Data Factory has three groupings of activities: data movement activities, data transformation activities, and control activities. An activity can take zero or more input datasets and produce one or more output datasets. The following diagram shows the relationship between pipeline, activity, and dataset in Data Factory:

An input dataset represents the input for an activity in the pipeline, and an output dataset represents the output for the activity. Datasets identify data within different data stores, such as tables, files, folders, and documents. After you create a dataset, you can use it with activities in a pipeline. For example, a dataset can be an input/output dataset of a Copy Activity or an HDInsightHive Activity.

https://docs.microsoft.com/en-us/azure/data-factory/concepts-pipelines-activities

Question 77: Skipped

When queries are submitted, a dedicated SQL pool query optimizer tries to determine which access paths to the data will result in the least amount of effort to retrieve the data required to resolve the query. It is a cost-based optimizer, and compares the cost of various query plans, and then chooses the plan with the lowest cost.

True or False: It is important to update the statistics after you load data or update large ranges of data, so that queries can benefit from the updated statistics information.

- False
- True
(Correct)

Explanation
When queries are submitted, a dedicated SQL pool query optimizer tries to determine which access paths to the data will result in the least amount of effort to retrieve the data required to resolve the query. It is a cost-based optimizer, and compares the cost of various query plans, and then chooses the plan with the lowest cost.

Statistics in dedicated SQL pools

To aid this process, statistics are required that describe the amount of data that is present within ranges of values, and range of rows that may be returned to fulfill a query filter or join. Therefore, after loading data into a dedicated SQL pool, collecting statistics on your data is one of the most important things you can do for query optimization.

When you create a database in a dedicated SQL pool in Azure Synapse Analytics, the automatic creation of statistics is turned on by default. This means that statistics are created when you run the following type of Transact-SQL statements:

- `SELECT`
- `INSERT-SELECT`
- `CTAS`
- `UPDATE`
- `DELETE`
- `EXPLAIN` when containing a join or the presence of a predicate is detected

When executing the above Transact-SQL statements, that the statistics creation is performed on the fly, and as a result, there can be a slight degradation in query performance.

To avoid this, statistics are also created on any index that you create that helps aid the query optimize process. As this is an action that is performed in advance of querying the table on which the index is based, it means that the statistics are created in advance. However, you must consider that as new data is loaded into the table, the statistics may become out of date.

As such, **it is important to update the statistics after you load data or update large ranges of data, so that queries can benefit from the updated statistics information.**

You can check if your data warehouse has `AUTO_CREATE_STATISTICS` configured by running the following command:

```
SQL

SELECT name, is_auto_create_stats_on

FROM sys.databases

If your data warehouse doesn't have AUTO_CREATE_STATISTICS enabled, it is recomme
nded that you enable this property by running the following command:

SQL

ALTER DATABASE <yourdatawarehousename>

SET AUTO_CREATE_STATISTICS ON
```

Statistics in serverless SQL pools

Statistics in a serverless SQL pool has the same objective of using a cost-based optimizer to choose an execution plan that will execute the fastest. How it creates its statistics is different.

Serverless SQL pool analyses incoming user queries for missing statistics. If statistics are missing, the query optimizer creates statistics on individual columns in the query predicate or join condition to improve cardinality estimates for the query plan. The `SELECT` statement will trigger automatic creation of statistics. You can also manually create statistics, this is important when working with CSV files, as automatic statistics creation is not enabled for them.

In the following example, a system stored procedure is used to specify the creation of statistics for a specific Transact-SQL statement

```
SQL

sys.sp_create_openrowset_statistics [ @stmt = ] N'statement_text'

To create statistics for a specific column within a csv file, you can run the fol
lowing code:

SQL

/* make sure you have the credentials to access the storage account created

IF EXISTS (SELECT * FROM sys.credentials WHERE name = 'https://azureopendatastora
ge.blob.core.windows.net/censusdatacontainer')

DROP CREDENTIAL [https://azureopendatastorage.blob.core.windows.net/censusdatacon
tainer]

GO


CREATE CREDENTIAL [https://azureopendatastorage.blob.core.windows.net/censusdatac
ontainer]

WITH IDENTITY='SHARED ACCESS SIGNATURE',

SECRET = ''

GO

*/


/*
```

The following code will create statistics on a column named year, from a file named `population.csv`

```
*/


EXEC sys.sp_create_openrowset_statistics N'SELECT year
FROM OPENROWSET(
BULK ''https://sqlondemandstorage.blob.core.windows.net/csv/population/population
.csv'',
FORMAT = ''CSV'',
FIELDTERMINATOR ='','',
ROWTERMINATOR = ''\n''
)
WITH (
[country_code] VARCHAR (5) COLLATE Latin1_General_BIN2,
[country_name] VARCHAR (100) COLLATE Latin1_General_BIN2,
[year] smallint,
[population] bigint
) AS [r]
‘
```

You should also update the statistics when the data in the files change. In fact, Serverless SQL pool automatically recreates statistics if data is changed significantly. Every time statistics are automatically created, the current state of the dataset is also saved: file paths, sizes, last modification dates.

To update statistics for the year column in the dataset, which is based on the population.csv file, you need to drop and then create them, here is the drop statement:

```
SQL
EXEC sys.sp_drop_openrowset_statistics N'SELECT year
FROM OPENROWSET(
BULK ''https://sqlondemandstorage.blob.core.windows.net/csv/population/population
.csv'',
FORMAT = ''CSV'',
FIELDTERMINATOR ='','',
ROWTERMINATOR = ''\n''
)
WITH (
```

```
[country_code] VARCHAR (5) COLLATE Latin1_General_BIN2,
[country_name] VARCHAR (100) COLLATE Latin1_General_BIN2,
[year] smallint,
[population] bigint
) AS [r]
‘
```

To update statistics for a statement, you need to drop and create statistics. The following stored procedure is used to drop statistics against a specific Transact-SQL text:

```
SQL
sys.sp_drop_openrowset_statistics [ @stmt = ] N'statement_text’
```

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql/develop-tables-statistics

Question 78: Skipped

Azure Data Factory is composed of four core components. These components work together to provide the platform on which you can compose data-driven workflows with steps to move and transform data.

Which component is best described by:

“It is created to perform a specific task by composing the different activities in the task in a single workflow. This can be scheduled to execute, or a trigger can be defined that determines when an execution needs to be kicked off.”

- Linked service
- Dataset
- Pipeline **(Correct)**
- Activity

Explanation

An Azure subscription might have one or more Azure Data Factory instances. Azure Data Factory is composed of four core components. These components work together to provide the platform on which you can compose data-driven workflows with steps to move and transform data.

• **Pipeline:** It is created to perform a specific task by composing the different activities in the task in a single workflow. Activities in the pipeline can be data ingestion (Copy data to Azure) -> data processing (Perform Hive Query). Using pipeline as a single task user can schedule the task and manage all the activities in a single process also it is used to run the multiple operation parallel. Multiple activities can be logically grouped together with an object referred to as a **Pipeline**, and these can be *scheduled* to execute, or a *trigger* can be defined that determines when a pipeline execution needs to be kicked off. There are different types of triggers for different types of events.

• **Activity:** It is a specific action performed on the data in a pipeline like the transformation or ingestion of the data. Each pipeline can have one or more activities in it. If the data is copied from one source to destination using Copy Monitor then it is a data movement activity. If data transformation is performed on the data using a hive query or spark job then it is a data transformation activity.

• **Datasets:** It is basically collected data users required which are used as input for the ETL process. Datasets have different formats; they can be in JSON, CSV, ORC, or text format.

• **Linked services:** It has information on the different data sources and the data factory uses this information to connect to data originating sources. It is mainly used to locate the data stores in the machines and also represent the compute services for the activity to be executed like running spark jobs on spark clusters or running hive queries using the hive services from the cloud.

https://www.educba.com/azure-data-factory/

Question 79: Skipped

If you are performing analytics on the data, set up the storage account as an Azure Data Lake Storage Gen2 account by setting the Hierarchical Namespace option to which of the following?

- Disabled
- Auto-scale
- OFF
- ON
- Enabled **(Correct)**

Explanation
In Azure Blob storage, you can store large amounts of unstructured ("object") data, in a single hierarchy, also known as a flat namespace. You can access this data by using HTTP or HTTPs. Azure Data Lake Storage Gen2 builds on blob storage and optimizes I/O of high-volume data by using hierarchical namespaces that you turned on in the previous exercise.

Hierarchical namespaces organize blob data into *directories* and stores metadata about each directory and the files within it. This structure allows operations, such as directory renames and deletes, to be performed in a single atomic operation. Flat namespaces, by contrast, require several operations proportionate to the number of objects in the structure. Hierarchical namespaces keep the data organized, which yields better storage and retrieval performance for an analytical use case and lowers the cost of analysis.

Azure Blob storage vs. Azure Data Lake Storage

If you want to store data *without performing analysis on the data*, set the **Hierarchical Namespace** option to **Disabled** to set up the storage account as an Azure Blob storage account. You can also use blob storage to archive rarely used data or to store website assets such as images and media.

If you are performing analytics on the data, set up the storage account as an Azure Data Lake Storage Gen2 account by setting the **Hierarchical Namespace** option to **Enabled**. Because Azure Data Lake Storage Gen2 is integrated into the Azure Storage platform, applications can use either the Blob APIs or the Azure Data Lake Storage Gen2 file system APIs to access data.

https://blog.pragmaticworks.com/azure-data-lake-vs-azure-blob-storage-in-data-warehousing

Question 80: Skipped

Which feature of Spark determines how your code is executed?

- Java Garbage Collection
- Cluster Configuration
- Tungsten Record Format
- Catalyst Optimizer **(Correct)**

Explanation
Spark SQL uses Catalyst's general tree transformation framework in four phases - Analysis, Logical Optimization, Physical Planning, and Code Generation.

Because the Databricks API is declarative, a large number of optimizations are available to us.

Some of the examples include:

• Optimizing data type for storage

• Rewriting queries for performance

• Predicate push downs

Among the most powerful components of Spark are Spark SQL. At its core lies the Catalyst optimizer. This extensible query optimizer supports both rule-based and cost-based optimization.

When you execute code, Spark SQL uses Catalyst's general tree transformation framework in four phases, as shown below:

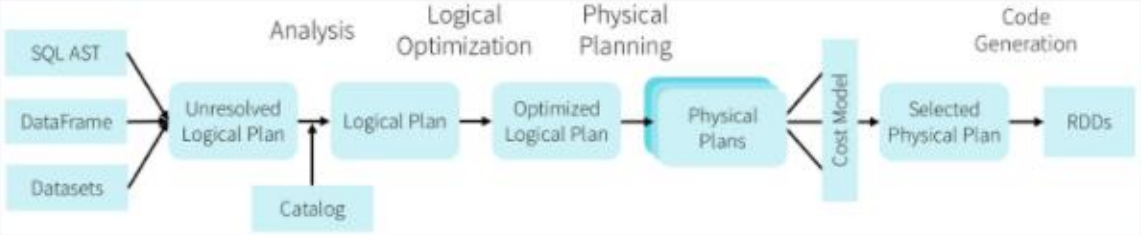
1. analyzing a logical plan to resolve references

2. logical plan optimization

3. physical planning

4. code generation to compile parts of the query to Java bytecode

In the physical planning phase, Catalyst may generate multiple plans and compare them based on cost. All other phases are purely rule-based.



Catalyst is based on functional programming constructs in Scala and designed with these key two purposes:

• Easily add new optimization techniques and features to Spark SQL

• Enable external developers to extend the optimizer (e.g. adding data source specific rules, support for new data types, etc.)

https://data-flair.training/blogs/spark-sql-optimization/

Question 81: Skipped

Scenario: You have been contracted by Wayne Enterprises, a company owned by Bruce Wayne with market value of over twenty seven million dollars. Bruce founded Wayne Enterprises shortly after he created the Wayne Foundation and he became the president and chairman of the company.

Bruce has come to you because his IT team plans to use Microsoft Azure Databricks and they are not as familiar with Azure as they would like to be. You have been hired as a consultant by Wayne Enterprises.

The plan is to create an Azure Databricks workspace that has a tiered structure. The workspace will contain the following three workloads:
• A workload for data engineers who will use Python and SQL.
• A workload for jobs that will run notebooks that use Python, Scala, and SOL.
• A workload that data scientists will use to perform ad hoc analysis in Scala and R.

The enterprise architecture team at Wayne Enterprises identifies the following standards for Databricks environments:
• The data engineers must share a cluster.

• The job cluster will be managed by using a request process whereby data scientists and data engineers provide packaged notebooks for deployment to the cluster.
• All the data scientists must be assigned their own cluster that terminates automatically after 120 minutes of inactivity. Currently, there are three data scientists.

Required: The team needs to create the Databricks clusters for the workloads.

Solution: The team creates a Standard cluster for each data scientist, a High Concurrency cluster for the data engineers, and a Standard cluster for the jobs.

Does this meet the requirement?

- No
  (Correct)
- Yes

Explanation
The solution offered does not meet the requirement of "A workload for jobs that will run notebooks that use Python, Scala, and SOL". Scala is only supported by Standard

Standard clusters

Standard clusters are recommended for a single user. Standard clusters can run workloads developed in any language: Python, R, Scala, and SQL.

High Concurrency clusters

A High Concurrency cluster is a managed cloud resource. The key benefits of High Concurrency clusters are that they provide Apache Spark-native fine-grained sharing for maximum resource utilization and minimum query latencies.

High Concurrency clusters work only for SQL, Python, and R. The performance and security of High Concurrency clusters is provided by running user code in separate processes, which is not possible in Scala.

In addition, only High Concurrency clusters support table access control.

https://docs.microsoft.com/en-us/azure/databricks/clusters/configure

Question 82: Skipped

True or False: Concurrency and the allocation of resources across connected users are also a factor that can limit the load performance into Azure Synapse Analytics SQL pools.

To optimize the load execution operations, recommendations are to reduce or minimize the number of simultaneous load jobs that are running or assigning higher resource classes that reduce the number of active running tasks.

- False
- True
  (Correct)

Explanation
Concurrency and the allocation of resources across connected users are also a factor that can limit the load performance into Azure Synapse Analytics SQL pools.

SQL Pools have the concept of concurrency slots, which manage the allocation of memory to connected users. To optimize the load execution operations, consider reducing or minimizing the number of simultaneous load jobs that are running or assigning higher resource classes that reduce the number of active running tasks.

https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/memory-concurrency-limits

Question 83: Skipped

Scenario: O'Shaughnessy's is a fast food restaurant. The chain has stores nationwide and is rivalled by Big Belly Burgers. You have been hired by the company to advise on the implementation of Azure migrating from an on-prem datacentre.

The IT team is working on a project to implement a lambda architecture on Microsoft Azure using an open-source big data solution for the purpose of aggregating, processing and maintaining data. During testing it is noted that the analytical data store is performing below expectations and management has come to you with the following requirement specifications.

Requirements:

• The solution must provide data warehousing

• The solution must reduce ongoing management activities

• The solution must deliver SQL query responses under one second

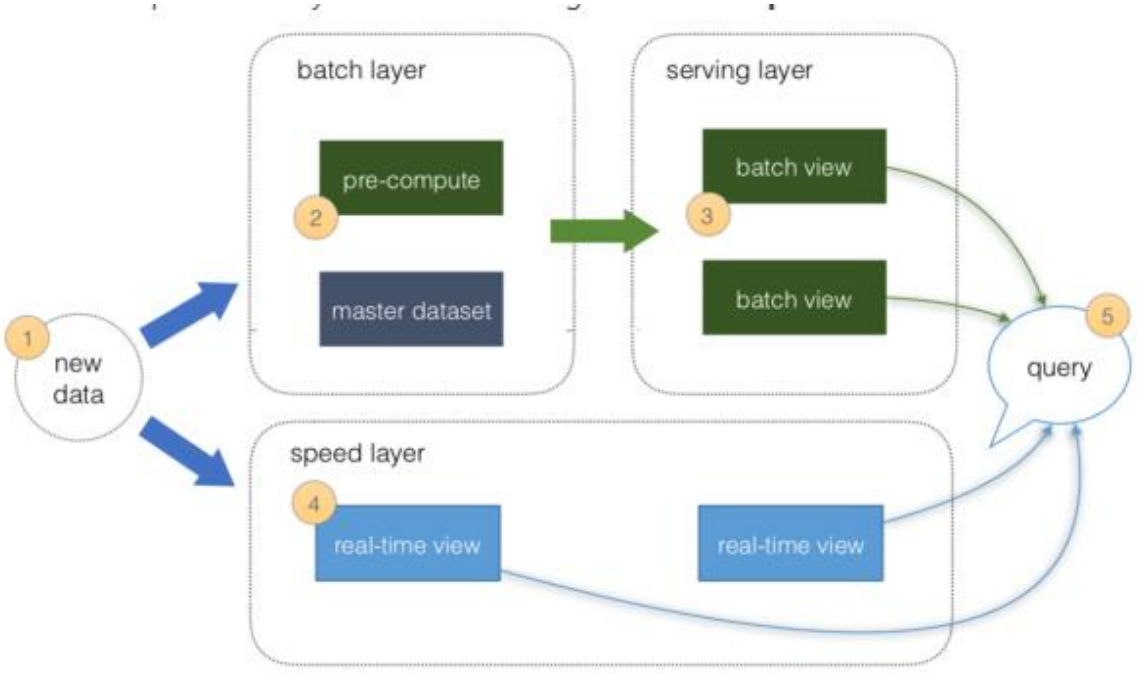
• The solution must create an HDInsight cluster to which fulfills all the listed requirements

As the expert consultant, the IT team is looking to you for direction. Which type of cluster should you advise them to create?

- Apache Hadoop
- Apache Spark
  (Correct)
- Interactive Query
- Apache HBase

Explanation
Lambda architecture is a data-processing architecture designed to handle massive quantities of data by taking advantage of both **batch** processing and **stream** processing methods, and minimizing the latency involved in querying **big data**.

It is a **Generic**, **Scalable**, and **Fault-tolerant** data processing architecture to address batch and speed latency scenarios with big data and **map-reduce**.



The system consists of three layers: Batch Layer, Speed Layer & Service Layer

1. All data is pushed into both the Batch layer and Speed layer.

2. The **Batch layer** has a master dataset (immutable, append-only set of raw data) and pre-computes the batch views.

3. The **Serving layer** has Batch views for fast queries.

4. The **Speed Layer** compensates for processing time (to the serving layer) and deals with recent data only.

5. All queries can be answered by merging results from Batch views and Real-time views or pinging them individually.

Lambda Architecture with Azure:

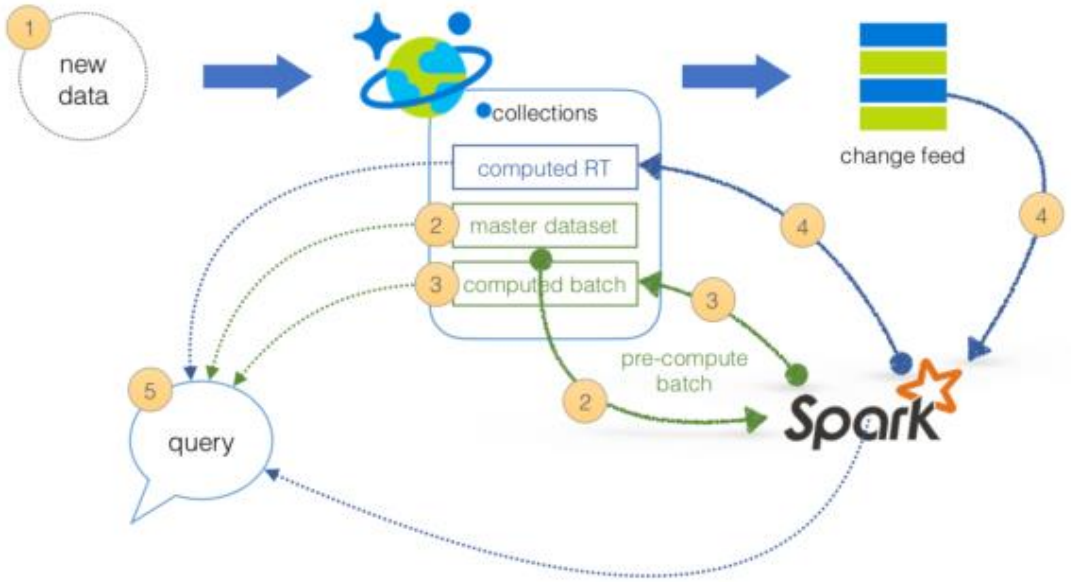
Azure offers you a combination of following technologies to accelerate real-time big data analytics:

1. Azure Cosmos DB, a globally distributed and multi-model database service.

2. Apache Spark for Azure HDInsight, a processing framework that runs large-scale data analytics applications.

3. Azure Cosmos DB change feed, which streams new data to the batch layer for HDInsight to process.

4. The Spark to Azure Cosmos DB Connector



How Azure simplifies the Lambda Architecture:

1. All data is pushed into **Azure Cosmos DB** for processing.

2. The **Batch layer** has a master dataset (immutable, append-only set of raw data) stored in Azure Cosmos DB. Using **HDI Spark**, you can pre-compute your aggregations to be stored in your computed **Batch Views**.

3. The **Serving layer** is an Azure Cosmos DB database with collections for the master dataset and computed Batch View for fast queries.

4. The **Speed layer** compensates for processing time (to the serving layer) and deals with recent data only. It utilizes **HDI Spark** to read the Azure Cosmos DB change feed. This enables you to persist your data as well as to query and process it concurrently.

5. All queries can be answered by merging results from batch views and real-time views, or pinging them individually.

https://docs.microsoft.com/en-us/azure/cosmos-db/lambda-architecture

Question 84: Skipped

An Azure Stream Analytics job supports which of the following input types? (Select three)

- ☐ Azure IoT Hub
  (Correct)
- ☐ Azure Data Lake Storage
- ☐ Azure Event Hub
  (Correct)
- ☐ Azure Table Storage
- ☐ Azure Queue Storage
- ☐ Azure Blob Storage
  (Correct)

Explanation
In Azure Stream Analytics, a *job* is a unit of execution. A Stream Analytics job pipeline consists of three parts:

• An **input** that provides the source of the data stream.

• A **transformation query** that acts on the input. For example, a transformation query could aggregate the data.

• An **output** that identifies the destination of the transformed data.

The Stream Analytics pipeline provides a transformed data flow from input to output, as the following diagram shows.

An Azure Stream Analytics job supports three input types:

• **Azure Event Hub**

Azure Event Hub consumes live streaming data from applications with low latency and high throughput.

• **Azure IoT Hub**

Azure IoT Hub consumes live streaming events from IoT devices. This service enables bi-directional communication scenarios where commands can be sent back to IoT devices to trigger specific actions based on analyzing streams they send to the service.

• **Azure Blob Storage**

Azure Blob Storage is used as the input source to consume files persisted in blob storage.

https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-monitoring

Question 85: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

When working with large data sets, it can take a long time to run the sort of queries that clients need. These queries can't be performed in real time, and often require algorithms such as MapReduce that operate in parallel across the entire data set. The results are then stored separately from the raw data and used for querying.

One drawback to this approach is that it introduces latency. If processing takes a few hours, a query may return results that are several hours old. Ideally, you would like to get some results in real time (perhaps with some loss of accuracy), and combine these results with the results from the batch analytics.

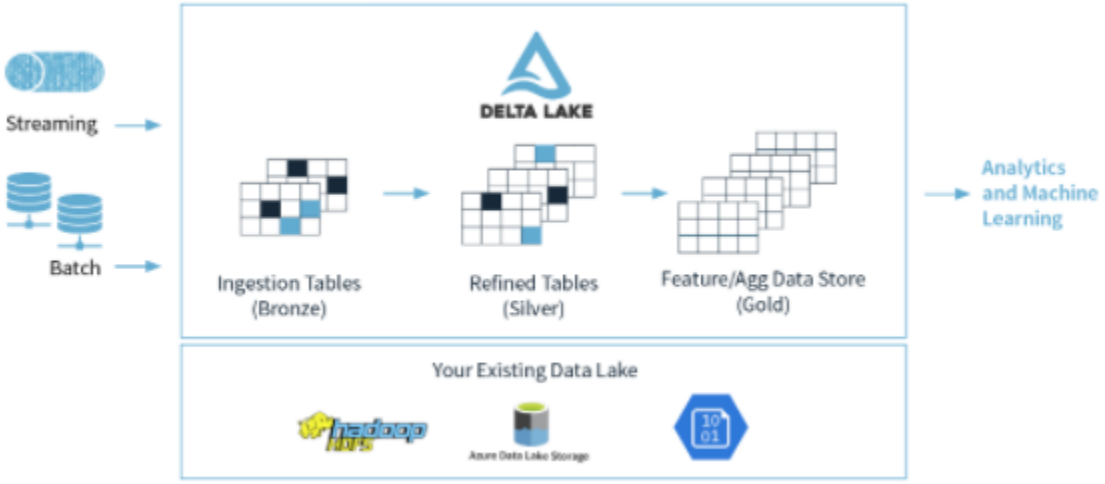
The Lambda architecture is a big data processing architecture that addresses this problem by combining both batch- and real-time processing methods. It features an append-only immutable data source that serves as system of record. Timestamped events are appended to existing events (nothing is overwritten). Data is implicitly ordered by time of arrival.

The [?] is a vast improvement upon the traditional Lambda architecture. At each stage, we enrich our data through a unified pipeline that allows us to combine batch and streaming workflows through a shared filestore with ACID-compliant transactions.

- No-SQL architecture
- Delta Lake architecture
  (Correct)
- Anaconda architecture
- Serverless architecture
- Data Lake architecture
- Data Sea architecture

Explanation

An example of a Delta Lake Architecture might be as shown in the diagram below.



• Many **devices** generate data across different ingestion paths.

• Streaming data can be ingested from **IoT Hub** or **Event Hub**.

• Batch data can be ingested by **Azure Data Factory** or **Azure Databricks**.

• Extracted, Transformed data is loaded into a **Delta Lake**.

Lambda architecture

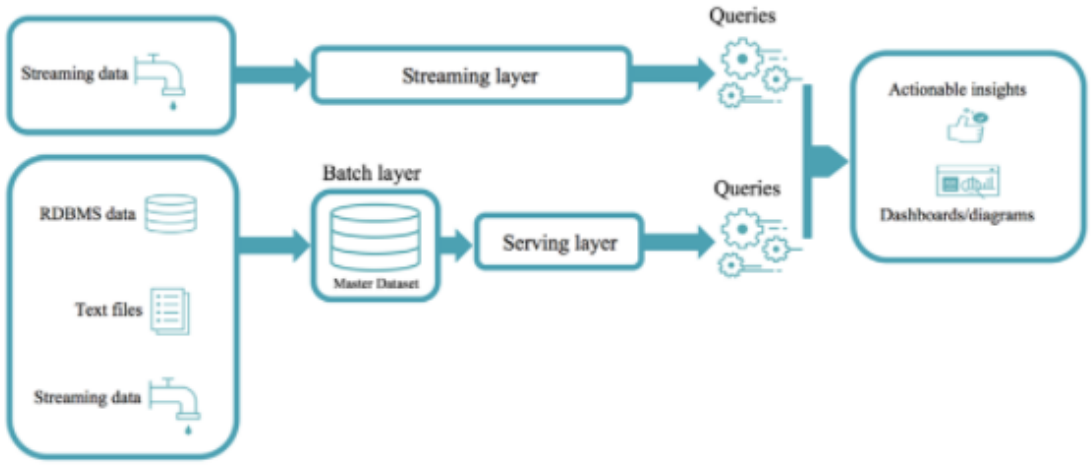
When working with large data sets, it can take a long time to run the sort of queries that clients need. These queries can't be performed in real time, and often require algorithms such as MapReduce that operate in parallel across the entire data set. The results are then stored separately from the raw data and used for querying.

One drawback to this approach is that it introduces latency. If processing takes a few hours, a query may return results that are several hours old. Ideally, you would like to get some results in real time (perhaps with some loss of accuracy), and combine these results with the results from the batch analytics.

The **lambda architecture** is a big data processing architecture that addresses this problem by combining both batch- and real-time processing methods. It features an append-only immutable data source that serves as system of record. Timestamped events are appended to existing events (nothing is overwritten). Data is implicitly ordered by time of arrival.

Notice how there are really two pipelines here, one batch and one streaming, hence the name *lambda* architecture.

It is difficult to combine processing of batch and real-time data as is evidenced by the diagram below:



Delta Lake architecture

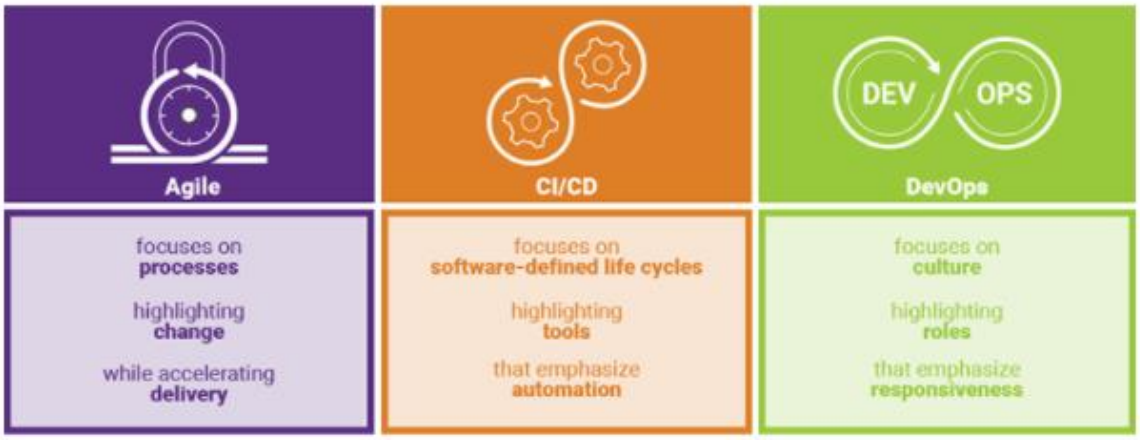
The Delta Lake Architecture is a vast improvement upon the traditional Lambda architecture. At each stage, we enrich our data through a unified pipeline that allows us to combine batch and streaming workflows through a shared filestore with ACID-compliant transactions.

Bronze tables contain raw data ingested from various sources (JSON files, RDBMS data, IoT data, etc.).

Silver tables will provide a more refined view of our data. We can join fields from various bronze tables to enrich streaming records, or update account statuses based on recent activity.

Gold tables provide business level aggregates often used for reporting and dashboarding. This would include aggregations such as daily active website users, weekly sales per store, or gross revenue per quarter by department.

The end outputs are actionable insights, dashboards, and reports of business metrics.



By considering our business logic at all steps of the extract-transform-load (ETL) pipeline, we can ensure that storage and compute costs are optimized by reducing unnecessary duplication of data and limiting ad hoc querying against full historic data.

Each stage can be configured as a batch or streaming job, and ACID transactions ensure that we succeed or fail completely.

https://www.jamesserra.com/archive/2019/10/databricks-delta-lake/

Question 86: Skipped

A DataFrame creates a data structure and it's one of the core data structures in Spark. In Spark, it is seen as a distributed collection of data that is organized into columns that have names.

Synapse Spark can be used to read and transform objects into a flat structure through data frames. Synapse SQL serverless can be used to query such objects directly and return those results as a regular table. With Synapse Spark, you can transform nested structures into columns and array elements into multiple rows.

The steps show the techniques involved to deal with complex data types have been shuffled.

a. Flatten nested schema Use the function to flatten the nested schema of the data frame (df) into a new data frame.

b. Define a function for flattening We define a function to flatten the nested schema.

c. Flatten child nested Schema Use the function you create to flatten the nested schema of the data frame into a new data frame.

d. Explode Arrays Transform the array in the data frame into a new dataframe where you also define the column that you want to select.

Which is the correct technique sequence to deal with complex data types?

- b → a → c → d
- a → c → b → d
- b → a → d → c
  (Correct)
- c → b → d → a

Explanation

A DataFrame creates a data structure and it's one of the core data structures in Spark. In Spark, it is seen as a distributed collection of data that is organized into columns that have names.

Some use cases for transforming complex data types are as follows:

• Complex data types are increasingly common and represent a challenge for data engineers as analyzing nested schema and arrays tend to include time-consuming and complex SQL queries.

• It can be difficult to rename or cast the nested columns data type.

• Performance issues arise when working with deeply nested objects.

• Data Engineers need to understand how to efficiently process complex data types and make them easily accessible to everyone.

Synapse Spark can be used to read and transform objects into a flat structure through data frames. Synapse SQL serverless can be used to query such objects directly and return those results as a regular table. With Synapse Spark, it's easy to transform nested structures into columns and array elements into multiple rows.

In the overview below, the steps show the techniques involved to deal with complex data types:

• Step 1: Define a function for flattening We define a function to flatten the nested schema.

• Step 2: Flatten nested schema Use the function to flatten the nested schema of the data frame (df) into a new data frame.

• Step 3: Explode Arrays Transform the array in the data frame into a new dataframe where you also define the column that you want to select.

• Step 4: Flatten child nested Schema Use the function you create to flatten the nested schema of the data frame into a new data frame.

https://medium.com/@saikrishna_55717/flattening-nested-data-json-xml-using-apache-spark-75fa4c8ea2a7

Question 87: Skipped

When working with Azure Data Factory, a dataset is a named view of data that simply points or references the data you want to use in your activities as inputs and outputs.

A dataset in Data Factory can be defined in a JSON format for programmatic creation as follows:

```
JSON
{
"name": "<name of dataset>",
"properties": {
"type": "<type of dataset: AzureBlob, AzureSql etc...>",
"linkedServiceName": {
"referenceName": "<name of linked service>",
"type": "LinkedServiceReference",
},
"schema": [
{
"name": "<Name of the column>",
"type": "<Name of the type>"
}
],
"typeProperties": {
"<type specific property>": "<value>",
"<type specific property 2>": "<value 2>",
}
}
}
```

Which of the JSON properties are required? (Select all that apply)

- ☐ structure
- ☐ type **(Correct)**
- ☐ typeProperties **(Correct)**
- ☐ name **(Correct)**

Explanation
When working with Azure Data Factory, a dataset is a named view of data that simply points or references the data you want to use in your activities as inputs and outputs.

Datasets identify data within different data stores, such as tables, files, folders, and documents. For example, an Azure Blob dataset specifies the blob container and folder in Blob storage from which the activity should read the data.

A dataset in Data Factory can be defined as an object within the Copy Data Activity, as a separate object, or in a JSON format for programmatic creation as follows:

```
JSON
{
"name": "<name of dataset>",
"properties": {
"type": "<type of dataset: AzureBlob, AzureSql etc...>",
"linkedServiceName": {
"referenceName": "<name of linked service>",
"type": "LinkedServiceReference",
},
"schema": [
{
"name": "<Name of the column>",
"type": "<Name of the type>"
}
],
"typeProperties": {
"<type specific property>": "<value>",
"<type specific property 2>": "<value 2>",
}
}
}
```

The following describes properties in the above JSON:

Property: name

Name of the dataset.

Required: Yes

Property: type

Type of the dataset. Specify one of the types supported by Data Factory (for example: AzureBlob, AzureSqlTable).

Required: Yes

Property: structure

Schema of the dataset.

Required: No

Property: typeProperties

The type properties are different for each type (for example: Azure Blob, Azure SQL table).

Required: Yes

https://docs.microsoft.com/en-us/azure/data-factory/v1/data-factory-create-datasets

Question 88: Skipped

Identify the missing word(s) in the following sentence within the context of Microsoft Azure.

Azure Data Factory is the cloud-based [?] and data integration service that allows you to create data-driven workflows for orchestrating data movement and transforming data at scale. You can build complex [?] processes that transform data visually with data flows or by using compute services such as Azure HDInsight Hadoop, Azure Databricks, and Azure Synapse Analytics.

- OLTP
- CI/CD
- OLAP

- ETL
  (Correct)
- ELT

Explanation
The need to trigger the batch movement of data, or to set up a regular schedule is a requirement for most analytics solutions. Azure Data Factory (ADF) is the service that can be used to fulfill such a requirement. ADF provides a cloud-based data integration service that orchestrates the movement and transformation of data between various data stores and compute resources.

Azure Data Factory is the cloud-based ETL (Extract, Transform, and Load) and data integration service that allows you to create data-driven workflows for orchestrating data movement and transforming data at scale. Using Azure Data Factory, you can create and schedule data-driven workflows (called pipelines) that can ingest data from disparate data stores. You can build complex ETL processes that transform data visually with data flows or by using compute services such as Azure HDInsight Hadoop, Azure Databricks, and Azure Synapse Analytics.

Much of the functionality of Azure Data Factory appears in Azure Synapse Analytics as a feature referred to as Pipelines, which enables you to integrate data pipelines between SQL Pools, Spark Pools and SQL Serverless, providing a one stop shop for all your analytical needs.

https://techcommunity.microsoft.com/t5/azure-data-factory/etl-in-the-cloud-is-made-easy-together-with-azure-data-factory/ba-p/1189736

Question 89: Skipped

Which of the following services allow customers to store semi-structured datasets in Azure.

- Azure Blob Storage
  (Correct)
- Azure File Storage
  (Correct)
- Azure Cosmos DB
  (Correct)
- Azure SQL Datawarehouse
- Azure Table Storage
  (Correct)
- Azure Content Delivery Network (CDN)
- Azure SQL for VM
- Azure SQL Database

Explanation
Azure Table Storage and Cosmos DB are both semi-structured (NoSQL) database services in Azure.

https://docs.microsoft.com/en-us/azure/cosmos-db/table-storage-overview

Azure Blob Storage and Azure File Storage are primarily for unstructured but we can store semi-structured data as well.

https://docs.microsoft.com/en-us/azure/search/search-semi-structured-data

Question 90: Skipped

Although you have the opportunity to ingest data at the source directly into a data warehouse, it is more typical to store the source data within a staging area, which is also referred to as a landing zone. This typically is a neutral storage area that sits between the source systems and the data warehouse.

Which are main reasons for adding a staging area into the architecture of a modern data warehouse? (Select all that apply)

- ☐ None of the listed options
- ☐ To rerun failed data warehouse loads from a staging area **(Correct)**
- ☐ All of the listed options
- ☐ Higher availability of data
- ☐ Data storage cost savings
- ☐ To join data together from different source systems **(Correct)**
- ☐ Enables you to deal with the ingestion of source systems on different schedules. **(Correct)**
- ☐ To reduce contention on source systems. **(Correct)**
- ☐ Quicker ETL/ELT processing times

Explanation

Although you have the opportunity to ingest data at the source directly into a data warehouse, it is more typical to store the source data within a staging area, which is also referred to as a landing zone. This typically is a neutral storage area that sits between the source systems and the data warehouse. The main reason for adding a staging area into the architecture of a modern data warehouse is for any one of the following reasons:

To reduce contention on source systems

Source systems typically play an important role in fulfilling business operations that either bring in revenue to an organization, or provides a function that is mission critical to the business. As a result, ingesting data from these systems must minimize the resource usage against the source system so it does not disrupt it. As a result, some data warehouse design strategies will involve grabbing data at a source, and “dumping” the data into a staging area.

This approach involves no transformation or cleansing. It simply grabs the data, so it minimizes the contention on the source system. This may also involve having the

source system output data into text files, that are then collected by your Extract, Transform and Load (ETL) process.

Enables you to deal with the ingestion of source systems on different schedules.

Staging environments provide a great place to store data from different source systems regardless of the schedule on which the data is ingested. For example, you may grab data from some source systems in the early evening because this is the time when they are at their quietest, and then it may not be until the early hours of the morning until you can grab data from other system as they have backup process running on them first before you are able to ingest the data. Having a staging area enables you to handle these different schedules

To join data together from different source systems

A staging environment provides the opportunity to bring together a single view of data from different source systems. As the staging area is independent from the source systems and the data warehouse, you have the freedom to perform any work you need without impacting these systems.

You can even create additional tables that can aid the process of joining data together from different source systems, referred to as mapping tables. In this scenario, imagine that you have a customer's table in one source system, that has a column named `FirstName`. In a second source system, perhaps running an AS400 system, you have customer's table that has a column named `FIRNAME` that also represents the first name of the customer too.

You can create a separate table that contains metadata that maps the data in a column from one source system, with another column from another source system that represent the same business entity. In this case `firstname`.

To rerun failed data warehouse loads from a staging area

Not all data warehouse loads will complete successfully, so your data warehouse has to be able to handle scenarios where a rerun of the ETL process may have to occur during core business hours, and needs to occur without disrupting the source systems again. By holding onto the staging data, you are able to rerun the ETL process from the staging area, rather than the source system.

In a modern data warehouse architecture, the source data can be so varied. The variety and volume of data that is generated and analyzed today is increasing. Companies have multiple sources of data, from websites to Point of Sale (POS) systems, and more recently from social media sites to Internet of Things (IoT) devices. Each source

provides an essential aspect of data that needs to be collected, analyzed, and potentially acted upon.

Based on this, Azure Data Lake Gen 2 is the ideal storage solution for hosting staging data as it contains a set of capabilities dedicated to big data analytics known as a data lake. A data lake is a repository of data that is stored in its natural format, usually as blobs or files. Azure Data Lake Storage is a comprehensive, scalable, and cost-effective data lake solution for big data analytics built into Azure.

Azure Data Lake Storage combines a file system with a storage platform to help you quickly identify insights into your data. Data Lake Storage Gen2 builds on Azure Blob storage capabilities to optimize it specifically for analytics workloads. This integration enables analytics performance, the tiering and data lifecycle management capabilities of Blob storage, and the high-availability, security, and durability capabilities of Azure Storage.

https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/modern-data-warehouse

Be prepared for the Microsoft