50+ Azure Data Factory Interview Questions and Answers [2023]

Ace Your Data Engineer Job Interview with a List of Top Azure Data Factory Interview Questions and Answers | ProjectPro
Last Updated: 12 May 2023

Discover 50+ Azure Data Factory interview questions and answers for all experience levels. These ADF interview questions and answers will help you demonstrate your expertise and impress your interviewer, increasing your chances of securing your ideal job.

A report by ResearchAndMarkets projects the global data integration market size to grow from USD 12.24 billion in 2020 to USD 24.84 billion by 2025, at a CAGR of 15.2% during the forecast period. This growth is due to the increasing adoption of cloud-based data integration solutions such as Azure Data Factory.

If you have heard about cloud computing, you would have heard about Microsoft Azure as one of the leading cloud service providers in the world, along with AWS and Google Cloud. As of 2023, Azure has ~23% of the cloud market share, second after AWS, and it is getting more popular daily. Azure Data Factory is one of the core services Microsoft Azure provides to work with data of any format or size at scale. Its intuitive, and data engineer-friendly interface helps anyone efficiently work with data at scale. The No-Code orchestration offered by Data Factory makes it an effective tool for any data engineer. The demand for data engineering will only grow as the data industry grows. For people looking to start a career in data engineering or advance in their career with data engineering, there is a high chance you may come across data engineer interview questions around ADF in your data engineer job interview.

So, prepare well and nail your data engineering interview with the most commonly asked Azure data factory questions and answers.

Table of Contents

Top Azure Data Factory Interview Questions and Answers in 2023

This list of Azure Data Factory interview questions and answers covers basic and experienced-level questions frequently asked in interviews, giving you a comprehensive understanding of the Azure Data Factory concepts. So, get ready to ace your interview with this complete list of Azure Data Factory interview questions and answers!

Azure Data Factory Interview Questions for Beginners

Below are the commonly asked interview questions for beginners on Azure Data Factory to help you ace your interview and showcase your skills and knowledge:

1. What is Azure Data Factory?

Azure Data Factory is a cloud-based, fully managed, serverless ETL and data integration service offered by Microsoft Azure for automating data movement from its native place to, say, a data lake or data warehouse using ETL (extract-transform-load) OR extract-load-transform (ELT). It lets you create and run data pipelines to help move and transform data and run scheduled pipelines.

2. Is Azure Data Factory ETL or ELT tool?

It is a cloud-based Microsoft tool that provides a cloud-based integration service for data analytics at scale and supports ETL and ELT paradigms.

3. Why is ADF needed?

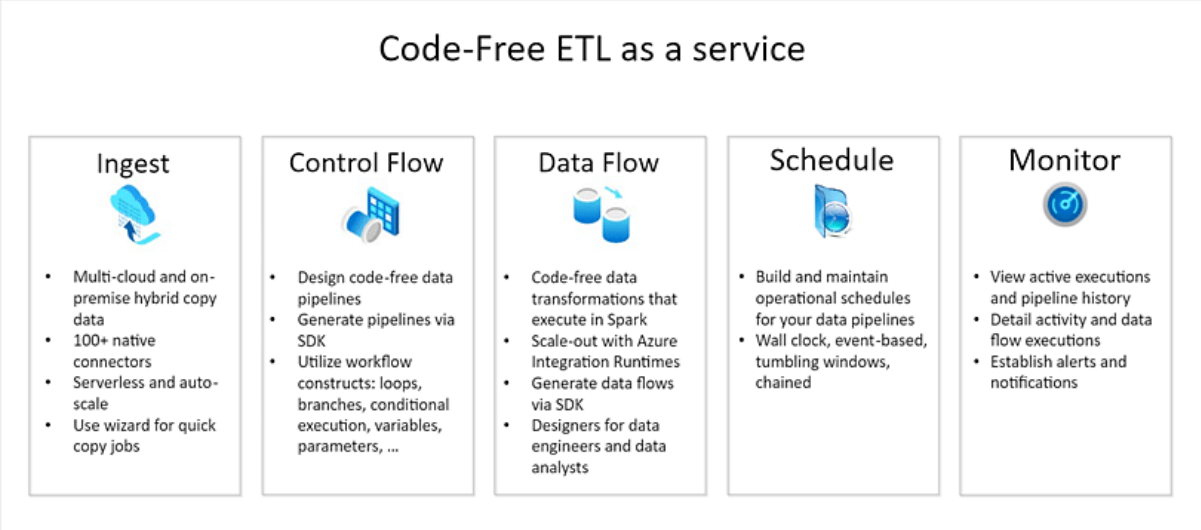
With an increasing amount of big data, there is a need for a service like ADF that can orchestrate and operationalize processes to refine the enormous stores of raw business data into actionable business insights.

4. What sets Azure Data Factory apart from conventional ETL tools?

Azure Data Factory stands out from other ETL tools as it provides: -

- Enterprise Readiness: Data integration at Cloud Scale for big data analytics!
- Enterprise Data Readiness: There are 90+ connectors supported to get your data from any disparate sources to the Azure cloud!
- Code-Free Transformation: UI-driven mapping dataflows.
- Ability to run Code on Any Azure Compute: Hands-on data transformations
- Ability to rehost on-prem services on Azure Cloud in 3 Steps: Many SSIS packages run on Azure cloud.
- Making DataOps seamless: with Source control, automated deploy & simple templates.
- Secure Data Integration: Managed virtual networks protect against data exfiltration, which, in turn, simplifies your networking.

Data Factory contains a series of interconnected systems that provide a complete end-to-end platform for data engineers. The below snippet summarizes the same.



New Projects

Build an ETL Pipeline with Talend for Export of Data from Cloud View Project

AWS CDK and IoT Core for Migrating IoT-Based Data to AWS View Project

End-to-End Snowflake Healthcare Analytics Project on AWS-1 View Project

A/B Testing Approach for Comparing Performance of ML Models View Project

Learn Efficient Multi-Source Data Processing with Talend ETL View Project

Learn to Create Delta Live Tables in Azure Databricks View Project

Multilabel Classification Project for Predicting Shipment Modes View Project

Azure Data Factory and Databricks End-to-End Project View Project

dbt Snowflake Project to Master dbt Fundamentals in Snowflake View Project

Build Serverless Pipeline using AWS CDK and Lambda in Python View Project

Build an ETL Pipeline with Talend for Export of Data from Cloud View Project

AWS CDK and IoT Core for Migrating IoT-Based Data to AWS View Project

End-to-End Snowflake Healthcare Analytics Project on AWS-1 View Project

A/B Testing Approach for Comparing Performance of ML Models View Project

Learn Efficient Multi-Source Data Processing with Talend ETL View Project

Learn to Create Delta Live Tables in Azure Databricks View Project

Multilabel Classification Project for Predicting Shipment Modes View Project

Azure Data Factory and Databricks End-to-End Project View Project

dbt Snowflake Project to Master dbt Fundamentals in Snowflake View Project

Build Serverless Pipeline using AWS CDK and Lambda in Python View Project

View all New Projects

5. What are the major components of a Data Factory?

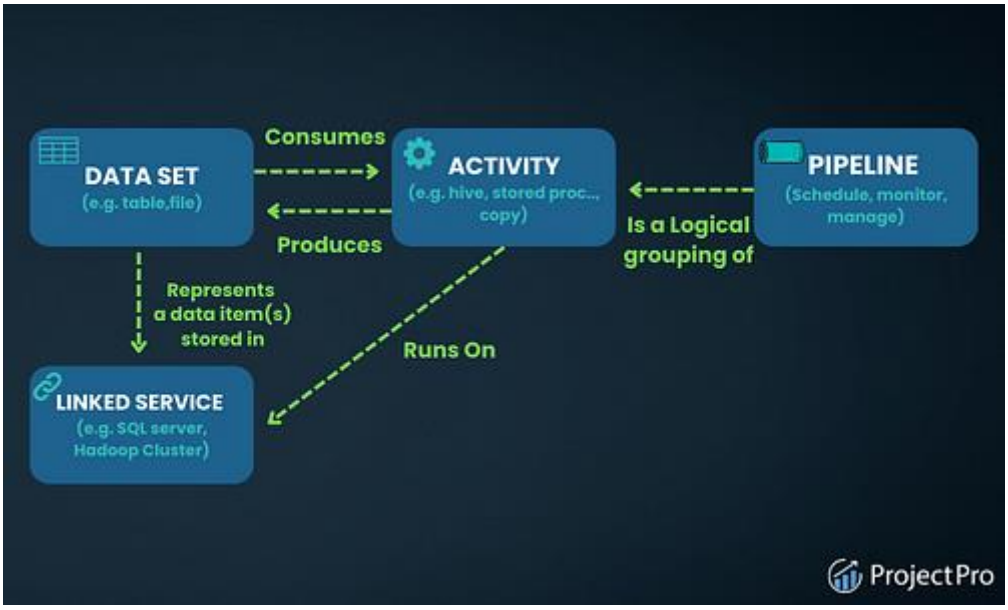
To work with Data Factory effectively, one must be aware of below concepts/components associated with it: -

- Pipelines: Data Factory can contain one or more pipelines, which is a logical grouping of tasks/activities to perform a task. An activity can read data from Azure blob storage and load it into Cosmos DB or Synapse DB for analytics while transforming the data according to business logic. This way, one can work with a set of activities using one entity rather than dealing with several tasks individually.

- Activities: Activities represent a processing step in a pipeline. For example, you might use a copy activity to copy data between data stores. Data Factory supports data movement, transformations, and control activities.
- Datasets: Datasets represent data structures within the data stores, which simply point to or reference the data you want to use in your activities as inputs or outputs.
- Linked Service: This is more like a connection string, which will hold the information that Data Factory can connect to various sources. In the case of reading from Azure Blob storage, the storage-linked service will specify the connection string to connect to the blob, and the Azure blob dataset will select the container and folder containing the data.
- Integration Runtime: Integration runtime instances bridged the activity and linked Service. The linked Service or activity references it and provides the computing environment where the activity runs or gets dispatched. This way, the activity can be performed in the region closest to the target data stores or compute Services in the most performant way while meeting security (no publicly exposing data) and compliance needs.
- Data Flows: These are objects you build visually in Data Factory, which transform data at scale on backend Spark services. You do not need to understand programming or Spark internals. Design your data transformation intent using graphs (Mapping) or spreadsheets (Power query activity).

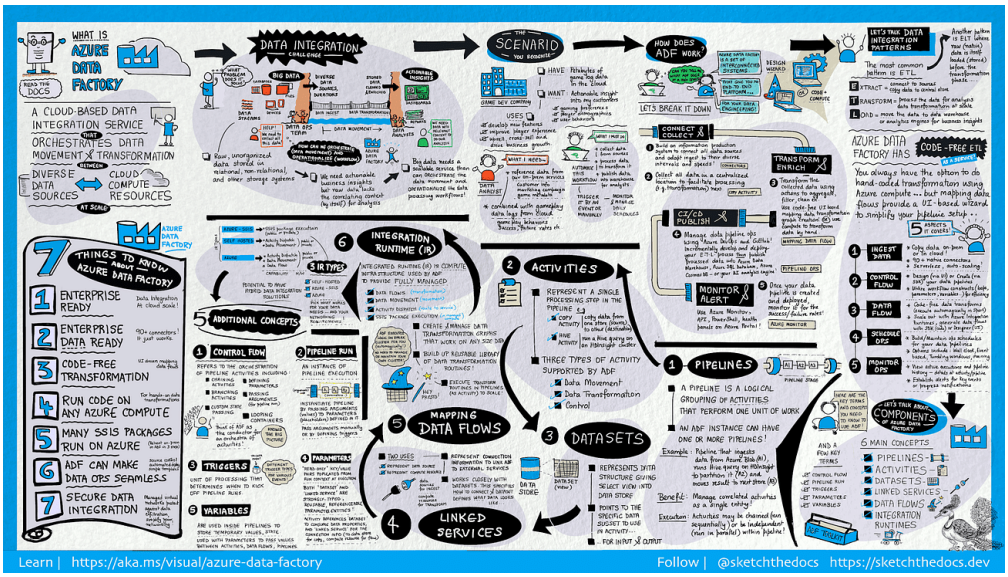
Refer to the documentation for more details: https://docs.microsoft.com/en-us/azure/data-factory/frequently-asked-questions

The below snapshot explains the relationship between pipeline, activity, dataset, and linked service.



You can also check:

https://docs.microsoft.com/en-us/azure/data-factory/media/introduction/data-factory-visual-guide.png



6. What are the different ways to execute pipelines in Azure Data Factory?

There are three ways in which we can execute a pipeline in Data Factory:

- Debug mode can be helpful when trying out pipeline code and acts as a tool to test and troubleshoot our code.
- Manual Execution is what we do by clicking on the 'Trigger now' option in a pipeline. This is useful if you want to run your pipelines on an ad-hoc basis.
- We can schedule our pipelines at predefined times and intervals via a Trigger. As we will see later in this article, there are three types of triggers available in Data Factory.

7. What is the purpose of Linked services in Azure Data Factory?

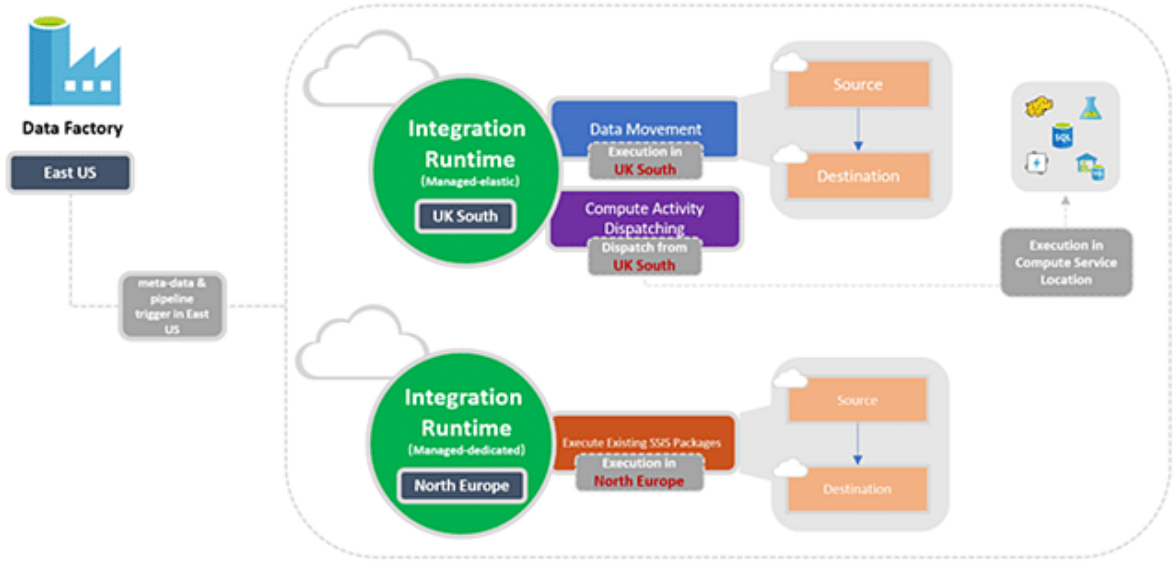
Linked services are used majorly for two purposes in Data Factory:

1. For a Data Store representation, i.e., any storage system like Azure Blob storage account, a file share, or an Oracle DB/ SQL Server instance.
2. For Compute representation, i.e., the underlying VM will execute the activity defined in the pipeline.

8. Can you Elaborate more on Data Factory Integration Runtime?

The Integration Runtime, or IR, is the compute infrastructure for Azure Data Factory pipelines. It is the bridge between activities and linked services. The linked Service or Activity references it and provides the computing environment where the activity is run directly or dispatched. This allows the activity to be performed in the closest region to the target data stores or computing Services.

The following diagram shows the location settings for Data Factory and its integration runtimes:



Source: https://docs.microsoft.com/en-us/azure/data-factory/concepts-integration-runtime

Azure Data Factory supports three types of integration runtime, and one should choose based on their data integration capabilities and network environment requirements.

1. Azure Integration Runtime: To copy data between cloud data stores and send activity to various computing services such as SQL Server, Azure HDInsight, etc.
2. Self-Hosted Integration Runtime: Used for running copy activity between cloud data stores and data stores in private networks. Self-hosted integration runtime is software with the same code as the Azure Integration Runtime but installed on your local system or machine over a virtual network.
3. Azure SSIS Integration Runtime: You can run SSIS packages in a managed environment. So, when we lift and shift SSIS packages to the data factory, we use Azure SSIS Integration Runtime.

9. What is required to execute an SSIS package in Data Factory?

We must create an SSIS integration runtime and an SSISDB catalog hosted in the Azure SQL server database or Azure SQL-managed instance before executing an SSIS package.

10. What is the limit on the number of Integration Runtimes, if any?

Within a Data Factory, the default limit on any entities is set to *5000*, including pipelines, data sets, triggers, linked services, Private Endpoints, and integration runtimes. If required, one can create an online support ticket to raise the limit to a higher number.

Refer to the documentation for more details: https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/azure-subscription-service-limits#azure-data-factory-limits.

11. What are ARM Templates in Azure Data Factory? What are they used for?

An ARM template is a JSON (JavaScript Object Notation) file that defines the infrastructure and configuration for the data factory pipeline, including pipeline activities, linked services, datasets, etc. The template will contain essentially the same code as our pipeline.

ARM templates are helpful when we want to migrate our pipeline code to higher environments, say Production or Staging from Development, after we are convinced that the code is working correctly.

12. How can we deploy code to higher environments in Data Factory?

At a very high level, we can achieve this with the below set of steps:

- Create a feature branch that will store our code base.
- Create a pull request to merge the code after we're sure to the Dev branch.
- Publish the code from the dev to generate ARM templates.

- This can trigger an automated CI/CD DevOps pipeline to promote code to higher environments like Staging or Production.

13. Which three activities can you run in Microsoft Azure Data Factory?

Azure Data Factory supports three activities: data movement, transformation, and control activities.

- Data movement activities: As the name suggests, these activities help move data from one place to another.
  e.g., Copy Activity in Data Factory copies data from a source to a sink data store.
- Data transformation activities: These activities help transform the data while we load it into the data's target or destination.
  e.g., Stored Procedure, U-SQL, Azure Functions, etc.
- Control flow activities: Control (flow) activities help control the flow of any activity in a pipeline.

e.g., wait activity makes the pipeline wait for a specified time.

14. What are the two types of compute environments supported by Data Factory to execute the transform activities?

Below are the types of computing environments that Data Factory supports for executing transformation activities: -

i) On-Demand Computing Environment: This is a fully managed environment provided by ADF. This type of calculation creates a cluster to perform the transformation activity and automatically deletes it when the activity is complete.

ii) Bring Your Environment: In this environment, you can use ADF to manage your computing environment if you already have the infrastructure for on-premises services.

15. What are the steps involved in an ETL process?

The ETL (Extract, Transform, Load) process follows four main steps:

i) Connect and Collect: Connect to the data source/s and move data to local and crowdsource data storage.

ii) Data transformation using computing services such as HDInsight, Hadoop, Spark, etc.

iii) Publish: To load data into Azure data lake storage, Azure SQL data warehouse, Azure SQL databases, Azure Cosmos DB, etc.

iv)Monitor: Azure Data Factory has built-in support for pipeline monitoring via Azure Monitor, API, PowerShell, Azure Monitor logs, and health panels on the Azure portal.

16. If you want to use the output by executing a query, which activity shall you use?

Look-up activity can return the result of executing a query or stored procedure.

The output can be a singleton value or an array of attributes, which can be consumed in subsequent copy data activity, or any transformation or control flow activity like ForEach activity.

17. Can we pass parameters to a pipeline run?

Yes, parameters are a first-class, top-level concept in Data Factory. We can define parameters at the pipeline level and pass arguments as you execute the pipeline run on demand or using a trigger.

18. Have you used Execute Notebook activity in Data Factory? How to pass parameters to a notebook activity?

We can execute notebook activity to pass code to our databricks cluster. We can pass parameters to a notebook activity using the *baseParameters* property. If the parameters are not defined/ specified in the activity, default values from the notebook are executed.

19. What are some useful constructs available in Data Factory?

- parameter: Each activity within the pipeline can consume the parameter value passed to the pipeline and run with the *@parameter* construct.
- coalesce: We can use the *@coalesce* construct in the expressions to handle null values gracefully.

- activity: An activity output can be consumed in a subsequent activity with the *@activity* construct.

20. Can we push code and have CI/CD (Continuous Integration and Continuous Delivery) in ADF?

Data Factory fully supports CI/CD of your data pipelines using Azure DevOps and GitHub. This allows you to develop and deliver your ETL processes incrementally before publishing the finished product. After the raw data has been refined into a business-ready consumable form, load the data into Azure Data Warehouse or Azure SQL Azure Data Lake, Azure Cosmos DB, or whichever analytics engine your business uses can point to from their business intelligence tools.

---

21. What do you mean by variables in the Azure Data Factory?

Variables in the Azure Data Factory pipeline provide the functionality to hold the values. They are used for a similar reason as we use variables in any programming language and are available inside the pipeline.

Set variables and append variables are two activities used for setting or manipulating the values of the variables. There are two types of variables in a data factory: -

i) System variables: These are fixed variables from the Azure pipeline. For example, pipeline name, pipeline id, trigger name, etc. You need these to get the system information required in your use case.

ii) User variable: A user variable is declared manually in your code based on your pipeline logic.

22. What are mapping data flows?

Mapping data flows are visually designed data transformations in Azure Data Factory. Data flows allow data engineers to develop a graphical data transformation logic without writing code. The resulting data flows are executed as activities within Azure Data Factory pipelines that use scaled-out Apache Spark clusters. Data flow activities can be operationalized using Azure Data Factory scheduling, control flow, and monitoring capabilities.

Mapping data flows provides an entirely visual experience with no coding required. Data flows run on ADF-managed execution clusters for scaled-out data processing. Azure Data Factory manages all the code translation, path optimization, and execution of the data flow jobs.

23. What is copy activity in the Azure Data Factory?

Copy activity is one of the most popular and universally used activities in the Azure data factory. It is used for ETL or Lift and Shift, where you want to move the data from one data source to another. While you copy the data, you can also do the transformation; for example, you read the data from the TXT/CSV file, which contains 12 columns; however, while writing to your target data source, you want to keep only seven columns. You can transform it and send only the required columns to the destination data source.

24. Can you elaborate more on the Copy activity?

The copy activity performs the following steps at high-level:

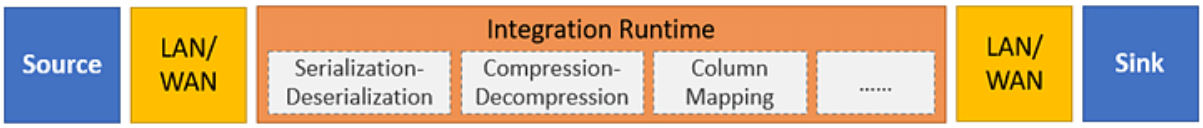
i) Read data from the source data store. (e.g., blob storage)

ii) Perform the following tasks on the data:

- Serialization/deserialization
- Compression/decompression
- Column mapping

iii) Write data to the destination data store or sink. (e.g., azure data lake)

This is summarized in the below graphic:



Source: https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-data-factory/3-how-azure-data-factory-works

Azure Data Factory Interview Questions for Experienced Professionals

Experienced professionals must understand its capabilities and features with the growing demand for ADF. Check out some of the most commonly asked Azure Data Factory interview questions for experienced professionals based on years of experience, providing insights into what employers are looking for and what you can expect in your next job interview.

Azure Data Factory Interview Questions For 3 Years Experience

Below are the most likely asked interview questions on Azure Data Factory for 3 years of experience professionals:

25. What are the different activities you have used in Azure Data Factory?

Here you can share some of the significant activities if you have used them in your career, whether your work or college project. Here are a few of the most used activities :

1. Copy Data Activity to copy the data between datasets.
2. ForEach Activity for looping.
3. Get Metadata Activity that can provide metadata about any data source.
4. Set Variable Activity to define and initiate variables within pipelines.
5. Lookup Activity to do a lookup to get some values from a table/file.
6. Wait Activity to wait for a specified amount of time before/in between the pipeline run.
7. Validation Activity will validate the presence of files within the dataset.
8. Web Activity to call a custom REST endpoint from an ADF pipeline.

26. How can I schedule a pipeline?

You can use the time window or scheduler trigger to schedule a pipeline. The trigger uses a wall-clock calendar schedule, which can schedule pipelines periodically or in calendar-based recurrent patterns (for example, on Mondays at 6:00 PM and Thursdays at 9:00 PM).

Currently, the service supports three types of triggers:

- Tumbling window trigger: A trigger that operates on a periodic interval while retaining a state.
- Schedule Trigger: A trigger that invokes a pipeline on a wall-clock schedule.
- Event-Based Trigger: A trigger that responds to an event. e.g., a file getting placed inside a blob.
  Pipelines and triggers have a many-to-many relationship (except for the tumbling window trigger). Multiple triggers can kick off a single pipeline, or a single trigger can kick off numerous pipelines.

27. When should you choose Azure Data Factory?

One should consider using Data Factory-

- When working with big data, there is a need for a data warehouse to be implemented; you might require a cloud-based integration solution like ADF for the same.
- Not all team members are experienced in coding and may prefer graphical tools to work with data.

- When raw business data is stored at diverse data sources, which can be on-prem and on the cloud, we would like to have one analytics solution like ADF to integrate them all in one place.
- We would like to use readily available data movement and processing solutions and be light regarding infrastructure management. So, a managed solution like ADF makes more sense in this case.

28. How can you access data using the other 90 dataset types in Data Factory?

The mapping data flow feature allows Azure SQL Database, Azure Synapse Analytics, delimited text files from Azure storage account or Azure Data Lake Storage Gen2, and Parquet files from blob storage or Data Lake Storage Gen2 natively for source and sink data source.

Use the Copy activity to stage data from any other connectors and then execute a Data Flow activity to transform data after it's been staged.

29. What is the difference between mapping and wrangling data flow (Power query activity)?

Mapping data flows transform data at scale without requiring coding. You can design a data transformation job in the data flow canvas by constructing a series of transformations. Start with any number of source transformations followed by data transformation steps. Complete your data flow with a sink to land your results in a destination. It is excellent at mapping and transforming data with known and unknown schemas in the sinks and sources.

Power Query Data Wrangling allows you to do agile data preparation and exploration using the Power Query Online mashup editor at scale via spark execution.

It supports 24 SQL data types from char, nchar to int, bigint and timestamp, xml, etc.

Refer to the documentation here for more details: https://docs.microsoft.com/en-us/azure/data-factory/frequently-asked-questions#supported-sql-types

Azure Data Factory Interview Questions For 4 Years Experience

If you're a professional with 4 years of experience in Azure Data Factory, check out the list of these common Azure Data Factory interview questions that you may encounter during your job interview.

30. Can a value be calculated for a new column from the existing column from mapping in ADF?

We can derive transformations in the mapping data flow to generate a new column based on our desired logic. We can create a new derived column or update an existing one when developing a derived one. Enter the name of the column you're making in the Column textbox.

You can use the column dropdown to override an existing column in your schema. Click the Enter expression textbox to start creating the derived column's expression. You can input or use the expression builder to build your logic.

31. How is the lookup activity useful in the Azure Data Factory?

In the ADF pipeline, the Lookup activity is commonly used for configuration lookup purposes, and the source dataset is available. Moreover, it retrieves the data from the source dataset and then sends it as the activity output. Generally, the output of the lookup activity is further used in the pipeline for making decisions or presenting any configuration as a result.

Simply put, lookup activity is used for data fetching in the ADF pipeline. The way you would use it entirely relies on your pipeline logic. Obtaining only the first row is possible, or you can retrieve the complete rows depending on your dataset or query.

32. Elaborate more on the Get Metadata activity in Azure Data Factory.

The Get Metadata activity is used to retrieve the metadata of any data in the Azure Data Factory or a Synapse pipeline. We can use the output from the Get Metadata activity in conditional expressions to perform validation or consume the metadata in subsequent activities.

It takes a dataset as input and returns metadata information as output. Currently, the following connectors and the corresponding retrievable metadata are supported. The maximum size of returned metadata is ***4 MB***.

Please refer to the snapshot below for supported metadata which can be retrieved using the Get Metadata activity.

Metadata options

You can specify the following metadata types in the Get Metadata activity field list to retrieve the corresponding information:

Metadata type	Description
itemName	Name of the file or folder.
itemType	Type of the file or folder. Returned value is `File` or `Folder`.
size	Size of the file, in bytes. Applicable only to files.
created	Created datetime of the file or folder.
lastModified	Last modified datetime of the file or folder.
childItems	List of subfolders and files in the given folder. Applicable only to folders. Returned value is a list of the name and type of each child item.
contentMD5	MD5 of the file. Applicable only to files.
structure	Data structure of the file or relational database table. Returned value is a list of column names and column types.
columnCount	Number of columns in the file or relational table.
exists	Whether a file, folder, or table exists. If `exists` is specified in the Get Metadata field list, the activity won't fail even if the file, folder, or table doesn't exist. Instead, `exists: false` is returned in the output.

Source: https://docs.microsoft.com/en-us/azure/data-factory/control-flow-get-metadata-activity#metadata-options

33. How to debug an ADF pipeline?

Debugging is one of the crucial aspects of any coding-related activity needed to test the code for any issues it might have. It also provides an option to debug the pipeline without executing it.

Azure Data Factory Interview Questions For 5 Years Experience

Here are some of the most likely asked Azure Data Factory interview questions for professionals with 5 years of experience to help you prepare for your next job interview and feel confident in showcasing your expertise.

34. What does it mean by the breakpoint in the ADF pipeline?

To understand better, for example, you are using three activities in the pipeline, and now you want to debug up to the second activity only. You can do this by placing the breakpoint at the second activity. To add a breakpoint, click the circle present at the top of the activity.

35. What is the use of the ADF Service?

ADF primarily organizes the data copying between relational and non-relational data sources hosted locally in data centers or the cloud. Moreover, you can use ADF Service to transform the ingested data to fulfill business requirements. In most Big Data solutions, ADF Service is used as an ETL or ELT tool for data ingestion.

36. Explain the data source in the Azure data factory.

The data source is the source or destination system that comprises the data intended to be utilized or executed. The data type can be binary, text, CSV, JSON, image files, video, audio, or a proper database.

Examples of data sources include Azure data lake storage, azure blob storage, or any other database such as MySQL DB, Azure SQL database, Postgres, etc.

37. Can you share any difficulties you faced while getting data from on-premises to Azure cloud using Data Factory?

One of the significant challenges we face while migrating from on-prem to the cloud is throughput and speed. When we try to copy the data using Copy activity from on-prem, the process rate could be faster, and hence we need to get the desired throughput.

There are some configuration options for a copy activity, which can help in tuning this process and can give desired results.

i) We should use the compression option to get the data in a compressed mode while loading from on-prem servers, which is then de-compressed while writing on the cloud storage.

ii) Staging area should be the first destination of our data after we have enabled the compression. The copy activity can decompress before writing it to the final cloud storage buckets.

iii) Degree of Copy Parallelism is another option to help improve the migration process. This is identical to having multiple threads processing data and can speed up the data copy process.

There is no right fit-for-all here, so we must try different numbers like 8, 16, or 32 to see which performs well.

iv) Data Integration Unit is loosely the number of CPUs used, and increasing it may improve the performance of the copy process.

38. How to copy multiple sheet data from an Excel file?

When using an Excel connector within a data factory, we must provide a sheet name from which we must load data. This approach is nuanced when we have to deal with a single or a handful of sheets of data, but when we have lots of sheets (say 10+), this may become a tedious task as we have to change the hard-coded sheet name every time!

However, we can use a data factory binary data format connector for this and point it to the Excel file and need not provide the sheet name/s. We'll be able to use copy activity to copy the data from all the sheets present in the file.

39. Is it possible to have nested looping in Azure Data Factory?

There is no direct support for nested looping in the data factory for any looping activity (for each / until). However, we can use one for each/until loop activity which will contain an execute pipeline activity that can have a loop activity. This way, when we call the looping activity, it will indirectly call another loop activity, and we'll be able to achieve nested looping.

Azure Data Factory Interview Questions for 6 Years Experience

Below are some of the most commonly asked Azure Data Factory advanced interview questions for professionals with 6 years of experience, helping to ensure that you are well-prepared for your next job interview.

40. How to copy multiple tables from one datastore to another datastore?

An efficient approach to complete this task would be:

- Maintain a lookup table/ file containing the list of tables and their source, which needs to be copied.
- Then, we can use the lookup activity and each loop activity to scan through the list.
- Inside the for each loop activity, we can use a copy activity or a mapping dataflow to copy multiple tables to the destination datastore.

41. What are some performance-tuning techniques for Mapping Data Flow activity?

We could consider the below set of parameters for tuning the performance of a Mapping Data Flow activity we have in a pipeline.

i) We should leverage partitioning in the source, sink, or transformation whenever possible. Microsoft, however, recommends using the default partition (size 128 MB) selected by the Data Factory as it intelligently chooses one based on our pipeline configuration.

Still, one should try out different partitions and see if they can have improved performance.

ii) We should not use a data flow activity for each loop activity. Instead, we have multiple files similar in structure and processing needs. In that case, we should use a wildcard path inside the data flow activity, enabling the processing of all the files within a folder.

iii) The recommended file format to use is '. parquet'. The reason being the pipeline will execute by spinning up spark clusters, and Parquet is the native file format for Apache Spark; thus, it will generally give good performance.

iv) Multiple logging modes are available: Basic, Verbose, and None.

We should only use verbose mode if essential, as it will log all the details about each operation the activity performs. e.g., It will log all the details of the operations performed for all our partitions. This one is useful when troubleshooting issues with the data flow.

The basic mode will give out all the necessary basic details in the log, so try to use this one whenever possible.

v) Try to break down a complex data flow activity into multiple data flow activities. Let's say we have several transformations between source and sink, and by adding more, we think the design has become complex. In this case, try to have it in multiple such activities, which will give two advantages:

- All activities will run on separate spark clusters, decreasing the run time for the whole task.
- The whole pipeline will be easy to understand and maintain in the future.

42. What are some of the limitations of ADF?

Azure Data Factory provides great functionalities for data movement and transformations. However, there are some limitations as well.

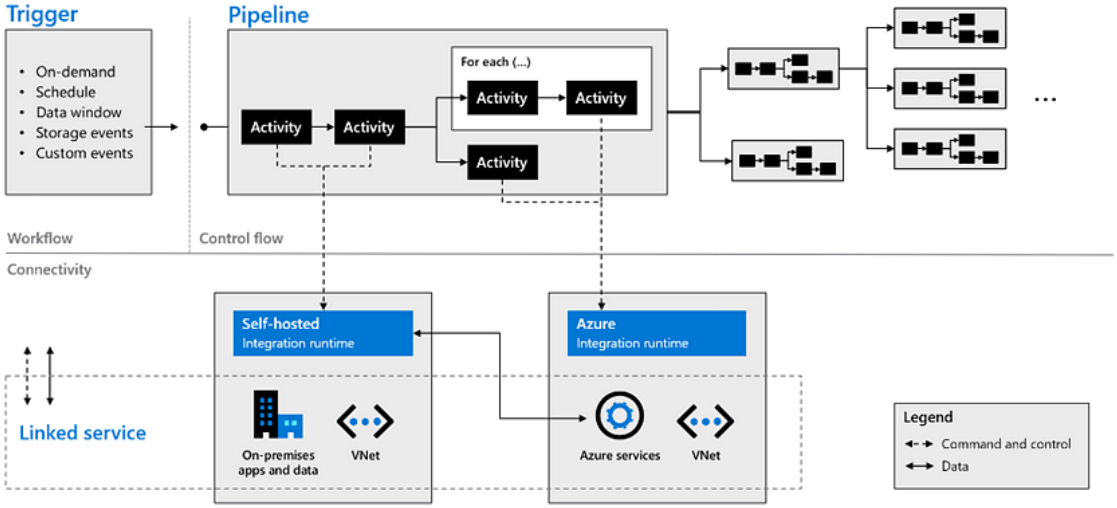
i) We can't have nested looping activities in the data factory, and we must use some workaround if we have that sort of structure in our pipeline. All the looping activities come under this: If, Foreach, switch, and until activities.

ii) The lookup activity can retrieve only 5000 rows at a time and not more than that. Again, we need to use some other loop activity along with SQL with the limit to achieve this sort of structure in the pipeline.

iii) We can have 40 activities in a single pipeline, including inner activity, containers, etc. To overcome this, we should modularize the pipelines regarding the number of datasets, activities, etc.

44. How are all the components of Azure Data Factory combined to complete an ADF task?

The below diagram depicts how all these components can be clubbed together to fulfill Azure Data Factory ADF tasks.



Source: https://docs.microsoft.com/en-us/learn/modules/intro-to-azure-data-factory/3-how-azure-data-factory-works

45. How do you send email notifications on pipeline failure?

There are multiple ways to do this:

1. Using Logic Apps with Web/Webhook activity.
   Configure a logic app that, upon getting an HTTP request, can send an email to the required set of people for failure. In the pipeline, configure the failure option to hit the URL generated by the logic app.
2. Using Alerts and Metrics from pipeline options.
   We can set up this from the pipeline itself, where we get numerous options for email on any activity failure within the pipeline.

46. Can we integrate Data Factory with Machine learning data?

Yes, we can train and retrain the model on machine learning data from the pipelines and publish it as a web service.

Checkout:https://docs.microsoft.com/en-us/azure/data-factory/transform-data-using-machine-learning#using-machine-learning-studio-classic-with-azure-data-factory-or-synapse-analytics

47. What is an Azure SQL database? Can you integrate it with Data Factory?

Part of the Azure SQL family, Azure SQL Database is an always up-to-date, fully managed relational database service built for the cloud for storing data. Using the Azure data factory, we can easily design data pipelines to read and write to SQL DB.

Checkout:https://docs.microsoft.com/en-us/azure/data-factory/connector-azure-sql-database?tabs=data-factory

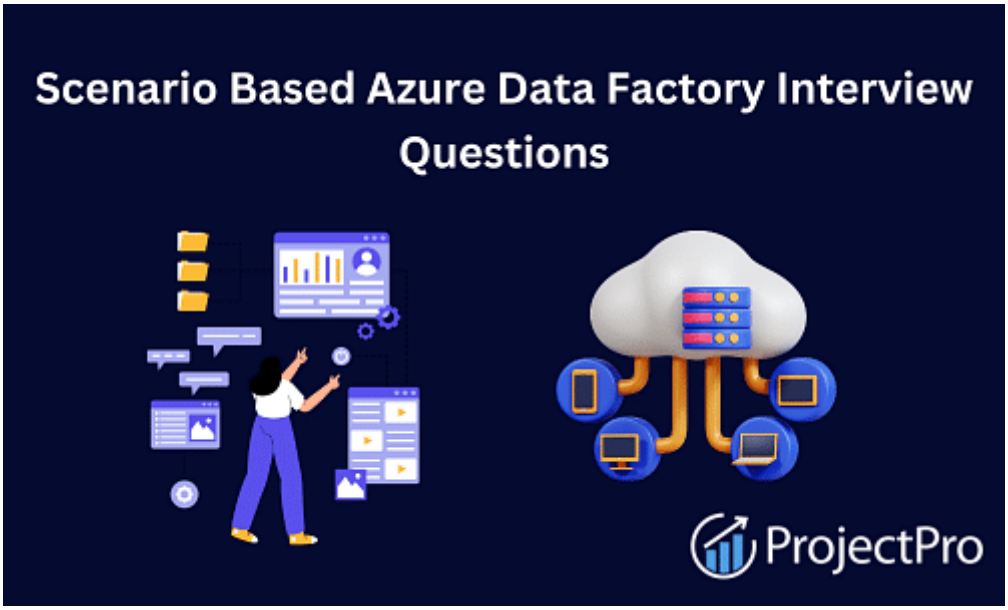
48. Can you host SQL Server instances on Azure?

Azure SQL Managed Instance is the intelligent, scalable cloud database service that combines the broadest SQL Server instance or SQL Server database engine compatibility with all the benefits of a fully managed and evergreen platform as a service.

50. What is Azure Data Lake Analytics?

Azure Data Lake Analytics is an on-demand analytics job service that simplifies storing data and processing big data.

Scenario-Based Azure Data Factory Interview Questions



If you are preparing for an interview for an Azure Data Factory role, it is essential to be familiar with various real-time scenarios that you may encounter on the job. Scenario-based interview questions are a popular way for interviewers to assess your

problem-solving abilities and practical knowledge of Azure Data Factory. Check out these common Azure data factory real-time scenario interview questions to help you prepare for your interview and feel more confident. So, let's dive in and discover some of the most commonly asked Azure Data Factory scenario-based interview questions below:

51. How would you set up a pipeline that extracts data from a REST API and loads it into an Azure SQL Database while managing authentication, rate limiting, and potential errors or timeouts during the data retrieval?

You can use the REST-linked Service to set up authentication and rate-limiting settings. To handle errors or timeouts, you can configure a Retry Policy in the pipeline and use Azure Functions or Azure Logic Apps to address any issues during the process.

52. Imagine merging data from multiple sources into a single table in an Azure SQL Database. How would you design a pipeline in Azure Data Factory to efficiently combine the data and ensure it is correctly matched and deduplicated?

You can use several strategies to efficiently merge and deduplicate data from multiple sources into a single table in an Azure SQL Database using Azure Data Factory. One possible approach involves using the Lookup and Join activities to combine data from different sources and the Deduplicate activity to remove duplicates. For performance optimization, you can use parallel processing by partitioning the data and processing each partition in parallel using the For Each activity. You can use a key column or set of columns to join and deduplicate the data to ensure that the data is correctly matched and deduplicated.

53. Imagine you must import data from many files stored in Azure Blob Storage into an Azure Synapse Analytics data warehouse. How would you design a pipeline in Azure Data Factory to efficiently process the files in parallel and minimize processing time?

Here is the list of steps that you can follow to create and design a pipeline in Azure Data Factory to efficiently process the files in parallel and minimize the processing time:

1. Start by creating a Blob storage dataset in Azure Data Factory to define the files' source location.
2. Create a Synapse Analytics dataset in Azure Data Factory to define the destination location in Synapse Analytics where the data will be stored.
3. Create a pipeline in Azure Data Factory that includes a copy activity to transfer data from the Blob Storage dataset to the Synapse Analytics dataset.
4. Configure the copy activity to use a binary file format and enable parallelism by setting the "parallelCopies" property.
5. You can also use Azure Data Factory's built-in monitoring and logging capabilities to track the pipeline's progress and diagnose any issues that may arise.

54. Suppose you work as a data engineer in a company that plans to migrate from on-premises infrastructure to Microsoft Azure cloud. As part of this migration, you intend to use Azure Data Factory (ADF) to copy data from a table in the on-premises Azure cloud. What actions should you take to ensure the successful execution of this pipeline?

One approach is to utilize a self-hosted integration runtime. This involves creating a self-hosted integration runtime that can connect to your on-premises servers.

55. Imagine you need to process streaming data in real time and store the results in an Azure Cosmos DB database. How would you design a pipeline in Azure Data Factory to efficiently handle the continuous data stream and ensure it is correctly stored and indexed in the destination database?

Here are the steps to design a pipeline in Azure Data Factory to efficiently handle streaming data and store it in an Azure Cosmos DB database.

1. Set up an Azure Event Hub or Azure IoT Hub as the data source to receive the streaming data.
2. Use Azure Stream Analytics to process and transform the data in real time using Stream Analytics queries.
3. Write the transformed data to a Cosmos DB collection as an output of the Stream Analytics job.
4. Optimize query performance by configuring appropriate indexing policies for the Cosmos DB collection.
5. Monitor the pipeline for issues using Azure Data Factory's monitoring and diagnostic features, such as alerts and logs.

ADF Interview Questions and Answers Asked at Top Companies

What questions do interviewers ask at top companies like TCS, Microsoft, or Mindtree? Check out these commonly asked data factory questions and answers to help you prepare.

TCS Azure Data Factory Interview Questions

Listed below are the most common Azure data factory interview questions asked at TCS:

56. How can one combine or merge several rows into one row in ADF? Can you explain the process?

In Azure Data Factory (ADF), you can merge or combine several rows into a single row using the "Aggregate" transformation.

57. How do you copy data as per file size in ADF?

You can copy data based on file size by using the "FileFilter" property in the Azure Data Factory. This property allows you to specify a file pattern to filter the files based on size.

Here are the steps you can follow to copy data based on the file size:

- Create a dataset for the source and destination data stores.
- Now, set the "FileFilter" property to filter the files based on their size in the source dataset.
- In the copy activity, select the source and destination datasets and configure the copy behavior per your requirement.
- Run the pipeline to copy the data based on the file size filter.

58. How can you insert folder name and file count from blob into SQL table?

You can follow these steps to insert a folder name and file count from blob into the SQL table:

- Create an ADF pipeline with a "Get Metadata" activity to retrieve the folder and file details from the blob storage.
- Add a "ForEach" activity to loop through each folder in the blob storage.
- Inside the "ForEach" activity, add a "Get Metadata" activity to retrieve the file count for each folder.
- Add a "Copy Data" activity to insert the folder name and file count into the SQL table.

- Configure the "Copy Data" activity to use the folder name and file count as source data and insert them into the appropriate columns in the SQL table.
- Run the ADF pipeline to insert the folder name and file count into the SQL table.

Microsoft Azure Data Factory Interview Questions

Below are the commonly asked ADF interview questions and answers asked at Microsoft:

59. Why do we require Azure Data Factory?

Azure Data Factory is a valuable tool that helps organizations simplify moving and transforming data between various sources and destinations, including on-premises data sources, cloud-based data stores, and software-as-a-service (SaaS) applications. It also provides a flexible and scalable platform for managing data pipelines, allowing users to create, schedule, and monitor complex data workflows easily. It also provides a variety of built-in connectors and integration options for popular data sources and destinations, such as Azure Blob Storage and Azure SQL Database.

60. Can you explain how ADF integrates with other Azure services, such as Azure Data Lake storage, Azure Blob Storage, and Azure SQL Database?

Azure Data Factory (ADF) can integrate with other Azure services such as Azure Data Lake Storage, Azure Blob Storage, and Azure SQL Database by using linked services. Linked services provide a way to specify the account name and credentials for the first two data sources to establish a secure connection. The 'Copy' activity can then transfer and transform data between these sources. With the 'Copy' activity, data can be moved between various source and sink data stores, including Azure Data Lake Storage, Azure Blob Storage, and Azure SQL Database. The 'Copy' activity also transforms the data while it is transferred.

End-to-End Snowflake Healthcare Analytics Project on AWS-1 View Project

PySpark Project-Build a Data Pipeline using Kafka and Redshift View Project

Build an AWS ETL Data Pipeline in Python on YouTube Data View Project

AWS Snowflake Data Pipeline Example using Kinesis and Airflow View Project

Snowflake Real Time Data Warehouse Project for Beginners-1 View Project

End-to-End Snowflake Healthcare Analytics Project on AWS-1 View Project

PySpark Project-Build a Data Pipeline using Kafka and Redshift View Project

Build an AWS ETL Data Pipeline in Python on YouTube Data View Project

AWS Snowflake Data Pipeline Example using Kinesis and Airflow View Project

Snowflake Real Time Data Warehouse Project for Beginners-1 View Project

End-to-End Snowflake Healthcare Analytics Project on AWS-1 View Project

View all Most Watched Projects

Mindtree Azure Data Factory Interview Questions

Here are a few commonly asked ADF questions asked at Mindtree:

62. What are the various types of loops in ADF?

Loops in Azure Data Factory are used to iterate over a collection of items to perform a specific action repeatedly. There are three major types of loops in Azure Data Factory:

- **For Each Loop**: This loop is used to iterate over a collection of items and perform a specific action for each item in the collection. For example, if you have a list of files in a folder and want to copy each file to another location, you can use a For Each Loop to iterate over the list of files and copy each file to the target location.
- **Until Loop:** This loop repeats a set of activities until a specific condition is met. For example, you could use an Until Loop to keep retrying an operation until it succeeds or until a certain number of attempts have been made.

- **While Loop:** This loop repeats a specific action while a condition is true. For example, if you want to keep processing data until a specific condition is met, you can use a While Loop to repeat the processing until the condition is no longer true.

63. Can you list all the activities that can be performed in ADF?

Here are some of the key activities that can be performed in ADF:

- Data Ingestion
- Data Transformation
- Data Integration
- Data Migration
- Data Integration
- Data Orchestration
- Data Enrichment

Here we shared top ADF interview questions and hope they will help you prepare for your next data engineer interview.

Master Your Data Engineering Skills with ProjectPro's Interactive Enterprise-Grade Projects

Whether a beginner or a seasoned professional, these interview questions on Azure Data Factory can help you prepare for your next Azure data engineering job interview. However, having practical experience is just as important, and therefore, you should focus on building your skills and expertise by gaining practical experience to flow in your career.

ProjectPro offers over 270+ solved end-to-end industry-grade projects based on big data and data science. These projects cover a wide range of industries and challenges, allowing you to build a diverse portfolio of work that showcases your skills to potential employers. So, what are you waiting for? Subscribe to ProjectPro Repository today to start mastering data engineering skills to the next level.

Access Data Science and Machine Learning Project Code Examples

FAQs on ADF Interview Questions

1. Is Azure Data Factory an ETL tool?

Yes, ADF is an ETL tool that helps to orchestrate and automate data integration workflows between data sources and destinations.

2. Which three types of activities can you run in Microsoft Azure Data Factory?

The three types of activities that you can run in the Azure factory are data movement activities, data transformation activities, and control activities.

3. What is the primary use of Azure Data Factory?

The primary use of Azure Data Factory is to help organizations manage their data integration workflows across various data sources and destinations. With Azure Data Factory, you can ingest data from multiple sources, transform and process it, and then load it into various destinations such as databases, data warehouses, and data lakes.