*Robert Jones*

*11/16/22*

**Azure Mini Project: End-to-end ETL**

*1. Why should one use Azure Key Vault when working in the Azure environment? What are the alternatives to using Azure Key Vault? What are the pros and cons of using Azure Key Vault?*

Azure Key Vault is useful when working in an environment where sensitive information such as passwords, certificates, API keys, and other information meant for select individuals should be kept in a secure, digital setting. Azure Key Vault also controls permissions for users via Azure Active Directory authentication. This means you can control which data is accessible by whom across many users. Alternatively, other cloud services offer their version of security. Traditionally, if a sensitive piece of code was needed, it could be stored separately in a controlled and secure environment. Pros include the scalability, support, and ease of use. Cons include stiff competition from other cloud providers, accessibility from Microsoft, mobile support, and no ability to connect with other software outside of Microsoft’s.

*2. How do you achieve the loop functionality within an Azure Data Factory pipeline? Why would you need to use this functionality in a data pipeline?*

A ForEach activity is reminiscent of a looping function in programming languages. If you’re looking to iterate over pipelines, find certain pipelines with filters and conditions, then execute those pipelines, this functionality will help you achieve that.

*3. What are expressions in Azure Data Factory? How are they helpful when designing a data pipeline (please explain with an example)?*

Expressions are JSON values that either evaluated at runtime or literal. They can be used literally and/or used with parameters to dynamically return a result based on an existing condition or other variable. For instance, an expression may be used to dynamically append dates to files, such as *@concat('Test\_', formatDateTime(utcnow(), 'yyyy-dd-MM'))*.

*4. What are the pros and cons of parametrizing a dataset in Azure Data Factory pipeline’s activity?*

Parameterization of a dataset allows for much more flexible ETL pipelines in ADF. It can be used to reduce the number of objects that a non-parameterized solution might use. By using a defined parameter, or using a predefined parameter on a dataset, we may indicate which data we’d like to reference to pass on to another step of a pipeline.

*5. What are the different supported file formats and compression codecs in Azure Data Factory? When will you use a Parquet file over an ORC file? Why would you choose an AVRO file format over a Parquet file format?*

Azure Data Factory supports Binary, text, Excel, JSON, ORC, Parquet, XML, and Avro.

Parquet vs ORC: Parquet is more capable of storing nested data. ORC is more capable of predicate pushdown. ORC supports Atomicity, Consistency, Isolation, and Durability (ACID) properties. ORC is more compression efficient.

Avro vs Parquet: Avro is row-based, while Parquet is column-based. Parquet is better for analytical queries. Writing in Avro is quicker than Parquet. Avro supports schema evolution, while Parquet only supports schema append. Parquet is good for querying a subset of columns, while Avro is good for ETL for all-column queries.