PDF 2: Data Processing Pipelines

Subjective Case Study Questions and Answers

Q1. ETL vs ELT Pipeline Design
Scenario: Process daily sales data from multiple sources into Azure Synapse.
Answer: For structured predictable data, ETL pipelines using ADF Mapping Data Flows extract, transform, and load. For large raw datasets, ELT pipelines load first, then transform inside Synapse. ETL ensures ready-to-use data; ELT preserves raw data and allows ad-hoc transformations.

Q2. Incremental Data Loads
Scenario: Load only new or changed records daily.
Answer: Use watermark columns to track last loaded timestamp. Copy Activity or Mapping Data Flow queries only new/changed records. This reduces data movement and keeps Synapse updated efficiently.

Q3. Handling Semi-Structured Data
Scenario: IoT sensors generate JSON files daily.
Answer: Connect to Blob Storage, use Mapping Data Flow to flatten JSON, transform fields, and load into Synapse tables. Schema drift handling and scheduling pipelines ensure automation.

Q4. Error Handling in Pipelines
Scenario: Mission-critical financial application requires reliability.
Answer: Implement retry policies, logging, alerts. Use Try-Catch or failure paths. Integrate with Azure Monitor. Design idempotent operations to prevent data corruption. Use checkpoints for long pipelines.

MCQs

1. Which ADF activity transforms data at scale?
   Answer: B. Mapping Data Flow — Spark-based transformations.
2. Best method for incremental loads?
   Answer: B. Watermark columns — track last processed records.
3. Handling nested JSON into structured tables?
   Answer: B. Mapping Data Flow + Flatten transformation.
4. Handling transient failures automatically?
   Answer: B. Retry policies and error handling activities.
5. Orchestration of multiple dependent pipelines?
   Answer: A. Execute Pipeline Activity — calls child pipelines.
6. Transformation to combine or group rows?
   Answer: B. Aggregate — performs grouping and calculations.
7. Trigger type for daily runs?
   Answer: A. Tumbling Window Trigger — recurring execution.
8. Reliability strategy for mission-critical pipelines?
   Answer: A. Idempotent design + retry policies — safe repeated runs.