Module - 2

Data Source:

Structured client transaction logs(ACID) - PostgresSQL --> good for structured transactional data

Portfolio performance reports - Azure Blob Storage --> cost effective and scalable for semi-structured files

SEC fillings - Azure Data Lake Storage -->Ideal for large, varied, regulatory documents

Module - 3
Elasticsearch for Unstructured Text
Provides fast full-text search and relevance scoring for research reports without upfront schema constraints.
Scales horizontally for large corpus of documents.
Integrates via REST API easily into Python/R pipelines.
PostgreSQL for Structured Ratings
ACID compliance ensures integrity of analyst ratings and relationships.
Mature ecosystem (ODBC, JDBC) for BI tools and Python/R connectors.
Fixed schema simplifies joins and aggregations.
Trade-off: less flexible for schema evolution but ratings schema is stable.
MongoDB for Alternative Data
Handles evolving, semi-structured data (shipping logs, image metadata) without schema migrations.
Dynamic document model supports varied fields per record.
Trade-off: eventual consistency and higher storage overhead vs. flexibility.
Snowflake as Central Data Warehouse
Consolidates curated data for analytics, ML, and reporting.
Supports ANSI SQL; integrates seamlessly with Python/R connectors.
Scales compute and storage independently, optimizing cost vs. performance.
Trade-off: data latency from micro-batch ingestion, which is acceptable for analytics.
Overall Trade-offs
Cost vs. Scalability: Leveraging cloud-native systems (Elasticsearch Service, Snowflake) incurs managed service fees but reduces operational overhead and ensures elastic scaling.
Flexibility vs. Consistency: Mix of NoSQL and RDBMS balances evolving data ingestion with

data integrity requirements.
Query Performance vs. Storage Efficiency: Specialized stores (Elasticsearch for search, Snowflake for analytics) optimize query performance at the cost of data duplication and ETL overhead.

Module 5

Data source:

-Trade reconcilation logs : Relational Database (SQL) -> As large data size and structural data

-Dark web scraping feeds: Graph like database(neo4j) to model relationship between user and post

-Employee access patterns: Time-based database ex)Timescale DB