# Initial Technical Interview

**Objective:** Design and implement an end-to-end data ingestion pipeline using a simplified telecoms dataset. This exercise will assess your proficiency in Data Engineering principles and your ability to address the needs of various internal user personas (Data Producers, Data Engineers, Data Consumers).

Have this running on a cloud-platform provider of choice (eg. GCP, Azure, AWS) using their free-tier. Bell will not provide compensation for any accidental cost overrun.   
I chose MS Azure as the cloud platform.

Focus on a practical, working solution. You will be assessed on your articulation of design choices made, and how they relate to the above internal user personas.

# Phase 1: Data Modeling and Generation

1. **Schema Design:** Design a relational database schema for a simplified telecoms dataset. This schema should include at least three tables (Customers, Usage, Billing), each with at least five relevant attributes and appropriate data types. Consider data normalization and potential future expansion.

Create an **Azure SQL Database** with three tables:

CREATE TABLE Customers (

customer\_id INT PRIMARY KEY IDENTITY(1,1),

name NVARCHAR(100),

email NVARCHAR(100) UNIQUE,

phone\_number NVARCHAR(20),

signup\_date DATE

);

CREATE TABLE Usage (

usage\_id INT PRIMARY KEY IDENTITY(1,1),

customer\_id INT,

call\_minutes INT,

data\_used\_MB FLOAT,

text\_messages INT,

usage\_date DATE,

FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id)

);

CREATE TABLE Billing (

billing\_id INT PRIMARY KEY IDENTITY(1,1),

customer\_id INT,

bill\_date DATE,

amount\_due DECIMAL(10,2),

payment\_status NVARCHAR(20) CHECK (payment\_status IN ('Paid', 'Unpaid', 'Pending')),

payment\_method NVARCHAR(50),

FOREIGN KEY (customer\_id) REFERENCES Customers(customer\_id)

);

1. **Data Generation:** Generate a synthetic dataset of approximately 10,000-20,000 rows conforming to your schema. The data should be realistic and representative of telecoms data, including variability and potential outliers.

Use faker and pandas to generate realistic telecom data:  
import pandas as pd  
import random  
from faker import Faker  
fake = Faker()  
# Define the number of rows  
num\_records = 10000  
# Generate Customers table  
customers = pd.DataFrame({  
 "customer\_id": range(1, num\_records + 1),  
 "name": [fake.name() for \_ in range(num\_records)],  
 "email": [fake.unique.email() for \_ in range(num\_records)],  
 "phone\_number": [fake.phone\_number() for \_ in range(num\_records)],  
 "signup\_date": [fake.date\_between(start\_date='-3y', end\_date='today') for \_ in range(num\_records)]})  
# Generate Usage table with unique customer\_id for each record  
usage = pd.DataFrame({  
 "usage\_id": range(1, num\_records + 1),  
 "customer\_id": customers["customer\_id"], # Ensures each customer has a unique usage entry  
 "call\_minutes": [random.randint(0, 600) for \_ in range(num\_records)], # Random call minutes per month  
 "data\_used\_MB": [round(random.uniform(0, 10000), 2) for \_ in range(num\_records)], # Random mobile data usage  
 "text\_messages": [random.randint(0, 500) for \_ in range(num\_records)], # Number of SMS sent  
 "usage\_date": [fake.date\_between(start\_date='-1y', end\_date='today') for \_ in range(num\_records)]})  
# Generate Billing table with unique customer\_id for each record  
billing = pd.DataFrame({  
 "billing\_id": range(1, num\_records + 1),  
 "customer\_id": customers["customer\_id"], # Ensures each customer has a billing entry  
 "bill\_date": [fake.date\_between(start\_date='-1y', end\_date='today') for \_ in range(num\_records)],  
 "amount\_due": [round(random.uniform(10, 300), 2) for \_ in range(num\_records)], # Monthly bill amount  
 "payment\_status": [random.choice(["Paid", "Unpaid", "Pending"]) for \_ in range(num\_records)],  
 "payment\_method": [random.choice(["Credit Card", "PayPal", "Bank Transfer", "Cash"]) for \_ in range(num\_records)]})  
# Save as CSV files  
customers.to\_csv("customers.csv", index=False)  
usage.to\_csv("usage.csv", index=False)

billing.to\_csv("billing.csv", index=False)  
print("Synthetic telecom dataset (10,000 rows per table) with unique customer\_id in usage and billing tables generated successfully!")

# Phase 2: Data Ingestion Pipeline

1. **Technology Selection:** Choose appropriate technologies for data extraction, transformation, loading (ETL), and storage. Your choices should consider factors like scalability, cost, ease of use, and maintainability. Be prepared to justify your technology choices during the interview, considering the needs of different personas.
2. **Pipeline Implementation:** Implement the ETL/ELT pipeline. This should include data extraction from the database, data transformation (cleaning, validation, enrichment, aggregation, etc.), and data loading into a chosen storage solution.
3. **Data Storage:** Select a suitable method for storing the processed data. Consider factors such as scalability, cost, and query performance.

# Phase 3: Data Visualization

1. **Visualization Selection:** Choose a suitable data visualization tool.
2. **Visualization Creation:** Create at least two visualizations that demonstrate insights from the ingested data. These visualizations should be clear, concise, and effectively communicate relevant information to a business audience.

# Phase 4: Persona Responsibilities

Define the key responsibilities for each of the following personas in relation to the data ingestion pipeline you have built:

* **Data Producers:** Describe the roles and responsibilities of the teams or individuals responsible for providing the raw data. What are their expectations regarding data quality, delivery mechanisms, and data governance?
* **Data Engineers:** Describe the roles and responsibilities of the data engineering team in maintaining and evolving the data ingestion pipeline. What are their key tasks and how do they interact with Data Producers and Data Consumers?
* **Data Consumers (Business Analysts, Data Scientists, etc.):** Describe the roles and responsibilities of the various data consumers who will utilize the processed data. What are their data needs, and how does the pipeline support their analytical requirements?

**Deliverables:**

* A fully functional end-to-end data ingestion pipeline.
* The synthetic telecoms dataset used.
* The code for the data ingestion pipeline, including comments explaining key design decisions.
* The visualizations created.

**How to succeed:** The interview will focus on a discussion of your:

* design choices
* technology selection
* data modeling decisions
* the rationale behind your pipeline implementation

Be prepared to discuss the scalability, maintainability, and cost-effectiveness of your solution, and how your design addresses the needs of different user personas:

* Data Producers
* Data Engineers/Operations
* Data Consumers

No formal architectural design documents or PowerPoint presentation are required. We’re expecting the coding and design to be done in 4 hours of work.