Data Engineering Exam

(The demo contains more exercises than the final exam!)

Name:

Code:

Q1: Data Modeling (11 points)

Design a star schema for an e-commerce system based on the following OLTP structure:

customers (id, name, email, phone, address)
orders (id, customer_id, order_date, status, total_amount)
products (id, name, description, price, stock_quantity)
order_items (id, order_id, product_id, quantity, unit_price, total_price)
shipments (id, order_id, carrier, tracking_number, shipment_date, delivery_date)

Requirements:

1. Identify the fact table and its measures.
2. Identify the dimension tables and their attributes.
3. Draw the star schema as a logical data model.
4. Ensure that appropriate dimension tables allow tracking historical changes.
5. Ensure the schema can answer the following example business questions:
    - What were the total sales by product category in the last quarter?
    - Who are the top customers by revenue generated?
    - Which carriers have the fastest average delivery times?
    - How many products are currently out of stock?

Q2 (5 points):

- Draw and label a diagram that includes the typical stages of the data engineering lifecycle.
- Write a one-sentence explanation of the goal for each stage.
- If applicable, provide one example tool or framework commonly used for each stage (e.g., those we saw in class).

Q3 (5 points):

Briefly describe the following methodology for data modelling:

- Entity-Relationship
- Dimensional Modelling
- Data Vault Storage

Q4. (3 points):

Match the following terms with their correct definitions:

Schema-on-read	High flexibility
	High data quality
	Best for analytical workloads
Schema-on-write	Best for transactional systems
	Low latency
	Requires schema during ingestion
	Applies schema at query time

Q5. (3 points):

Provide a detailed explanation of each stage in the ETL process.

Make sure to include examples for clarity.

Data Engineering Exam
(The demo contains more exercises than the final exam!)

Name:

Code:

Q6. Design Problem (11 points)

You are a data engineer at SuperShop, an e-commerce platform. Every day, your online store generates sales transactions that are stored across two databases in a distributed system. As part of the data analytics team, you've been tasked with creating a daily data pipeline that aggregates this sales data, after combining them with the daily purchase prices which are in a CSV file. As sales happen in different countries, all the prices shall be converted in euro before merge, the exchange rate can be taken from an web service called "ChangeMyCurrency". Also, if there is any mistakenly reported sale, e.g., with a nergative price, it should be removed and logged into an system for error handling. After the transformations are completed, and the pipeline loads data into a centralised data warehouse.

You are required to design the DAG of the pipeline as it would have been done in Airflow. Like in the practice, try to make it so that each node does the minimum amount of work as it is reasonable i.e don't delegate more than one task per node.

- Draw the pipeline at conceptual level (the general design of the pipeline)
    - Identify sources and sinks
    - Identify the transformation needed
    - Break transformation into tasks.
- Draw the pipeline at logical level (Airflow DAG)
- Describe the pipeline at physical level, i.e., hints to potential implementations of the operators.
- You need only the operators saw in class

Q7 Data Wrangling:

Pre-process the following table to reach the correct answer to the following SQL query calculating the salary and age average. List the necessary cleaning passages (e.g., noisy/missing data handling) and correct the data on the table.

Query	Avg Age	Avg Salary
SELECT AVG(Age) AS AverageAge, AVG(Salary) AS AverageSalary FROM Employees	35.2	1,220.00

Name	Age	Salary
John	32	1000
Sarah		1200
Dave	45	
Calment	222	900
Alice	38	800
Steve	29	1300
Dan		-1000
Riccardo	32	1800
Elon		10000000

Data Engineering Exam
(The demo contains more exercises than the final exam!)

Name:

Code:

Q8. (3 points):

Explain the reasons that led to the development of NoSQL databases, and provide a high-level description of the few families .

Bonus: position them wrt the CAP theorem

Bonus: position them within the data engineering pipeline