Web and Database Technologies – Q2 2022/23 – Midterm

CSE 1500 - Task Sheet

Important Information:

- **On this sheet you find the exam tasks. You write your responses on the accompanying answer sheet that has been distributed to you!**
- You may use the distributed scratch paper as you see fit, but do not use it for your final solutions.
- The overall time is **90 minutes**. Please briefly assess all tasks before starting and **prioritize the order** of your work per your strength and weaknesses!
- **After the exam time is up, stop writing and stay seated (!!!).** We will collect the exam at your place. **Please wait until we announce that all exams have been collected**, and only leave your place after that!
- The only external items you may use are pens.

Task	max
1 Conceptual Modelling	18
2.1 Convert Conceptual to Logical	12
2.2 Functional Dependencies & Logical	10
3.1 FDs & Normalization	(3x3=) 9
3.2 Normalization	7
4 SQL	27
Overall	83

1. Conceptual Modelling

(18 points) The Dutch railway company NS intends to improve the reliability of their trains by employing data-driven predictive maintenance. Sensors are attached to trains, and the sensor data is analyzed with AI and statistics to predict when a train part needs maintenance or repairs. Your job is to design a conceptual schema for a database to collect information about which sensors are available in a train, in which train part they are installed, and what they can measure.

Task: Create a **Conceptual EER schema** that satisfies the following requirements. Try to be as complete and efficient as possible. Provide all primary keys and cardinalities! Comment your most important design choices and explain which described <u>data requirements</u> or constraints could not be represented in your diagram (if any).

Scenario:

There are various sensors models in use, which come in different types based on what they measure. For example, some models are vibration sensors, sound sensors, light sensors, or voltage sensors. Overall, there are around 50 of such different types. More measurement types might be added to the system later. Some sensors even support multiple measurement types. For each measurement type a sensor model supports, the possible sampling rate (i.e., the measurements per second), measured unit and expected precision are recorded. For each measurement type, we also store a brief description. Each sensor model also has a unique model number and a manufacturer stored.

For example, the IR9000 sensor from IntelliRail is a combined light and sound loudness sensor model. It measures light with a sample rate of 100 Hz using Lux as a unit with a precision of +/- 2%. Alternatively, it can also sample light at a rate of 1000 Hz in Lux, but then only with a precision of +/- 5%. Sound loudness is only supported with a 100Hz sampling rate, using decibel and an expected precision of +/- 5%.

Individual sensors are installed in train parts. There are two types of train parts: wagons and engines. Each train part has a unique serial number, a manufacturer, a model number, and an optional nick name. For wagons, the maximum number of seats is also recorded. In addition, we store for each train part the date when it first entered service. Furthermore, we keep a maintenance log for engines and record each maintenance date with a brief description of what was maintained. Entries in the maintenance log for each individual engine have an automatically increasing identifier.
For example, the train engine 186 028, a NS186 model, was first put into service in 2016, and is nicknamed "Marleen" by operators. The two most recent maintenance operations (operation 6 and 7 for that engine) were updating the drive computer software, and replacing the rear axle, both on Dec 20 2016.

Individual sensors have a unique serial number. We also store the date of purchase for that sensor. They are of a specific sensor model and are installed into a specific train part. This information needs to be recorded, and the installation date is also being stored. An installed sensor will never be moved to a different train part.

Finally, trains have a train number (used to distinguish them), and a start and end station (e.g., "IC 2244" Amsterdam Centraal to Vlissingen – this train also passes through Delft!). Trains are composed of different train parts. However, this composition can change over time: as a hypothetical example, on Dec 12 2022, IC 2244 used two engines (VIRM8201 and VIRM8203) with 2 wagons (VIRW4001 and VIRW4017), while from Dec 05 to Dec 09, it used 2 other engines and 3 other wagons. On Dec 10 and 11, the train was cancelled due to a severe technical malfunction and got disassembled entirely. During this time, it had no train parts at all, and a very unreliable bus was used to replace parts of its route.

2. Logical Relational Schema

2.1. Convert Conceptual to Logical

(12 points) You are given a small ER model for which you are asked to design a logical relational schema. For each relation, specify attributes, primary keys, and foreign key constraints. You are allowed and encouraged to enrich your answer with comments that clarify the assumptions you made.

As a reminder, an example of logical relational schema notation is the following:

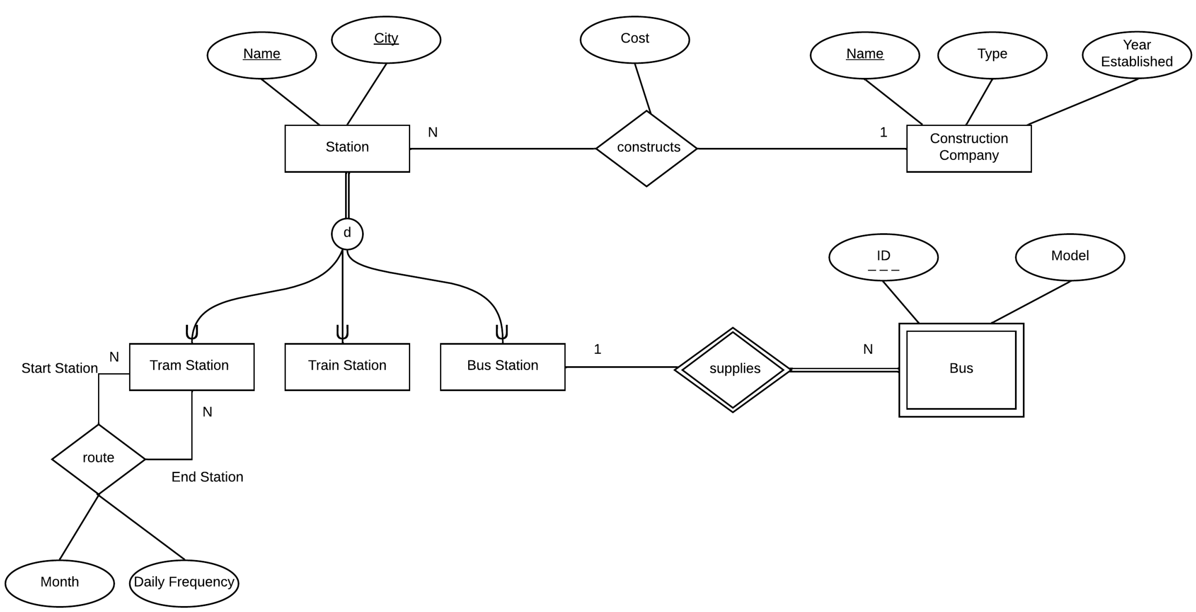
BOOK (<u>ID</u>, Title, Publisher -> PUBLISHER(Name))

PUBLISHER (<u>Name</u>, Address, Phone)

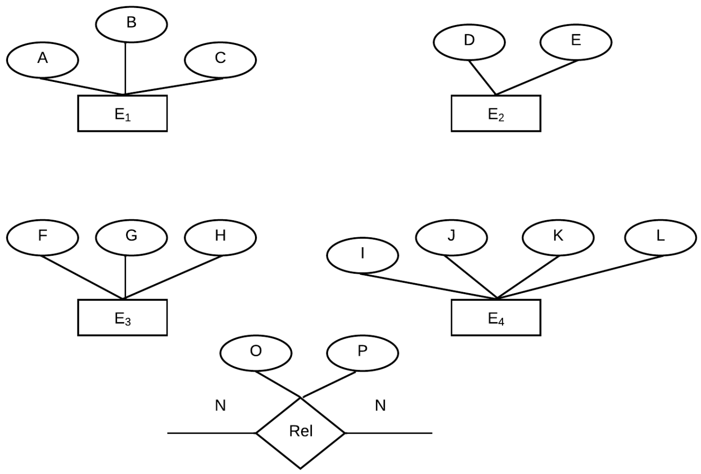
Task

You are given the following EER model about public transport stations. A station can be of three types: tram, train, or bus station. Moreover, we have information about start and end tram stations of routes whose daily frequency varies depending on the month!

Your task is to translate this conceptual EER model to a logical relational schema. Be aware that there might be multiple correct solutions, but you should opt for the one with **less data redundancy**!



2.2. Logical Relational Schema from Functional Dependencies



(10 points) You are given an incomplete part of an ER model which consists of four entities (E1, E2, E3, E4) with their corresponding attributes as shown in the figure above, and a many-to-many relationship (called "Rel") between a pair of entities (we do not have information which entities Rel connects). Information about the type of entities (strong or weak) and any other potential relationships (apart from "Rel") among them is missing. In addition, you are given the following five Functional Dependencies:

- $A \rightarrow \{B, C\}$
- $\{A, D\} \rightarrow E$
- $\{F, G, I, J\} \rightarrow K, L$
- $\{F, G\} \rightarrow H$
- $\{A, D, F, G, O\} \rightarrow P$

Task

Your task is to combine the information found in the functional dependencies with the incomplete conceptual schema, and derive and design a completed logical relational schema.

For each relation of that logical schema, specify attribute names, primary keys, and foreign key-primary key constraints. You can name the resulting relations as you wish (e.g., R_1, R_2, …).

As a reminder, an example of logical relational schema notation is the following:

BOOK (ID, Title, Publisher -> PUBLISHER(Name))

PUBLISHER (Name, Address, Phone)

3. Functional Dependencies and Normalization

3.1. Multiple-Choice: Functional Dependencies & Normalization

(9 points) Mark the correct answer for each sub-question in the **answer sheet**. (Only one choice is correct per question.)

1. Which of the following data instances do not violate any of the following two Functional Dependencies:
   i) {PassengerID, StartCity, EndCity } $\rightarrow$ Cost
   ii) TrainStationID $\rightarrow$ EndCity

   A.

PassengerID	TrainStationID	StartCity	EndCity	Cost
1	AB1	Amsterdam	Utrecht	12.3
2	AB1	Amsterdam	Utrecht	12.3

   B.

PassengerID	TrainStationID	StartCity	EndCity	Cost
1	AB1	Amsterdam	Utrecht	12.3
1	CD3	Amsterdam	Utrecht	12.8

   C.

PassengerID	TrainStationID	StartCity	EndCity	Cost
1	AB1	Amsterdam	Utrecht	12.3
2	AB1	Amsterdam	Dordrecht	15.5

   D. None of the above

2. Given the following FDs and the relation R(A,B,C,D,E), which answer applies?
   - A $\rightarrow$ B
   - {B, C} $\rightarrow$ D
   - D $\rightarrow$ E

   A. {A,B} is a candidate key.
   B. {A, B, C} is a superkey.
   C. A $\rightarrow$ B does not violate the 2NF definition.
   D. Relations R is in 2NF.

3. Given the following FDs and the relation R(A,B,C,D,E, F,G), which answer applies?
   - {B, C} $\rightarrow$ {G, F}
   - {B, F} $\rightarrow$ {G, C}
   - G $\rightarrow$ {A, D, E}

   A. {C, D, E, F, G} is a superkey.
   B. Relation R has only one candidate key.
   C. Relation R is not in 3NF.
   D. {B, C} cannot functionally determine all other attributes of R.

3.2. Normalization

(7 points) Suppose you are being given the following denormalized table that stores information about trains, operators of these trains, and routes between cities on specific dates that are operated by them.

1. Identify and list the Functional Dependencies that hold based only on the given data instances (**4 points**)
2. Normalize the table into 3NF using your functional dependencies from task 1. Show each resulting table with its instances. (you may shortcut Amsterdam as A, Delft as D, Brugge as BG, Brussels as BS, and Berlin as BE, etc.) (**3 points**)

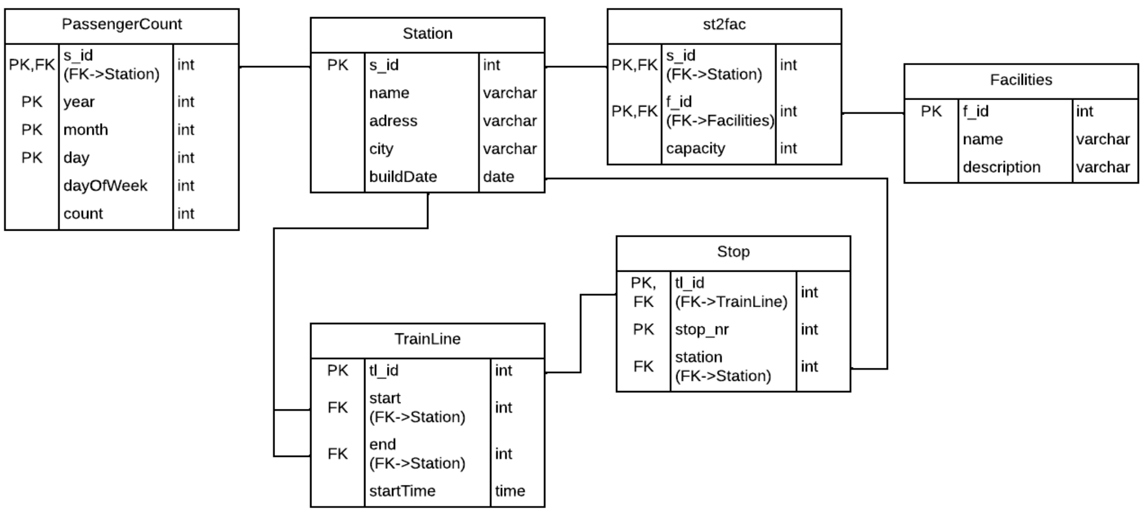
TrainID	TrainCompany	OperatorID	OperatorName	Route History (StartCity, EndCity, Date)
NS12	NS	A123	John Smith	(Amsterdam, Delft, 12/8/2021)
NS12	NS	B234	Mary Bell	(Amsterdam, Delft, 3/5/2022)
SNCB43	SNCB	AF11	Andrew Fall	(Brugge, Ghent, 3/2/2022)
SNCB43	SNCB	AF11	Andrew Fall	(Brussels, Liege, 1/9/2022)
SNCB43	SNCB	AF11	Andrew Fall	(Brugge, Liege, 8/10/2022)
SNCB43	SNCB	AF11	Andrew Fall	(Brussels, Ghent, 1/9/2022)
DB-1214	DB	SB-90	Sophie Braun	(Berlin, Koln, 11/8/2021)
DB-5432	DB	SB-90	Sophie Braun	(Frankfurt, Dusseldorf, 23/5/2022)
DB-5432	DB	SB-90	Sophie Braun	(Frankfurt, Dusseldorf, 13/8/2022)

4. SQL

The schema below describes Dutch railway stations (like Delft, Delft Campus, etc.), and their facilities. For example, station "Delft" has as facilities an OV-Fiets Point offering up to 75 rental bikes, and a GreenWheels Point offering up to 5 rental cars. It does not have luggage lockers. A city might have several stations, and station names are globally unique.

A passenger count is conducted every day using the automated gates, for example, there have been 12 000 passengers on 12.12.2022 (which was the first day of the week, a Monday).

Train Lines start at one station at a specific time, and end at another station. They have several stops in-between (with the start station being also the first stop). For example, IC 2252 starts 16:05 in Amsterdam Central towards Vlissingen. Delft is the 8th stop of that train line.



Task: Write SQL queries that fulfill the following tasks, based on the schema and information provided above.

1. List the names of all stations and their total passenger count for the entire year 2021. The schema of the result set should be (name, passenger_count) (**4 points**)
2. List all distinct pairs of names of start and end stations for train lines that have as 5th stop any station located in Rotterdam. The schema of the result set should be (Start_Station, End_Station). (**6 points**)
3. List the names of all facilities whose capacity across all stations in the database does not differ by more than 10. For example, the minimal capacity for "Wheel-Chair Enabled Toilets" is 0 in Arkel, Urk and some other stations, and 10 (which is the maximum) in Utrecht Centraal. Thus, "Wheel-Chair Enabled Toilets" fulfills the query criteria. The schema of the result set should be (facility_name) (**4 points**)
4. List for each train line that performs a circular route the number of unique train stations where it stops. The schema of the result set should be (tl_id, number_of_stops) (**4 points**)
5. List the average number of passengers that visited "Delft Station" during weekends of March 2022. The schema of the result set should be (avg_passenger_count) (**5 points**)
6. List all cities that have at least two different stations and less than 5 different types of facilities in total. For example, within the city of Delft, we have Station Delft with OV-Fiets & GreenWheels, and Delft Campus with OV-Fiets. These are 2 different types of facilities for the city of Delft across all stations. The schema of the result set should be (city). (**4 points**).