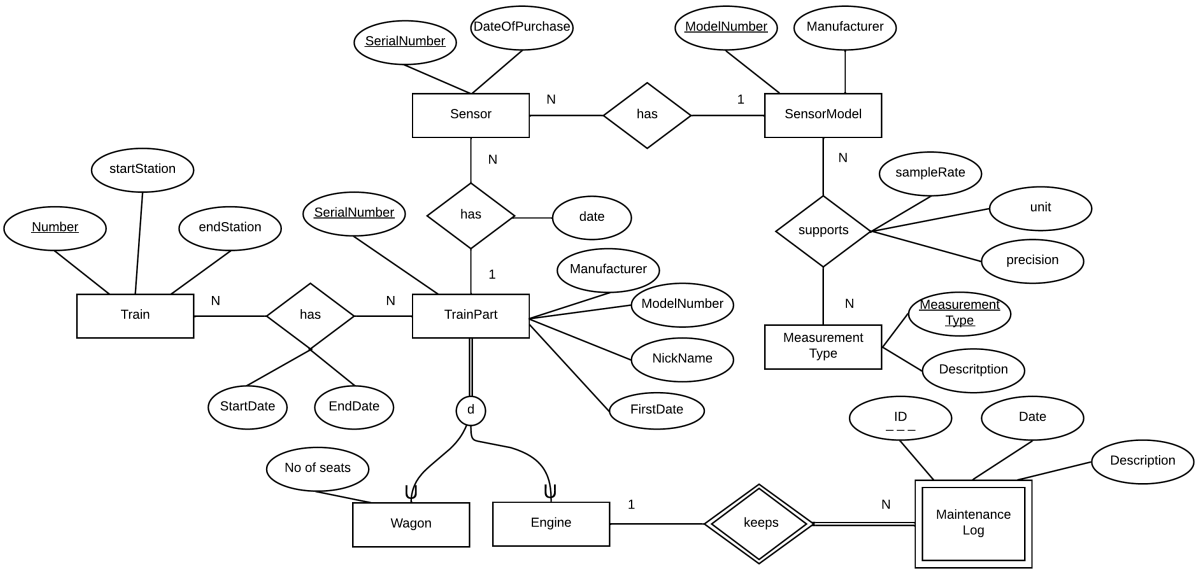
Web and Database Technologies – Q2 2022/23 – Midterm

CSE 1500 - Solutions

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



2. Logical Relational Schema

2.1. Convert Conceptual to Logical

1st Solution

Station (Name, City, Type, Cost, Construction_Company -> Construction Company(Name))

Construction Company (Name, Type, Year Established)

Bus (ID, Bus_Station -> Station(Name, City), Model)

Route (Start_Station -> Station(Name, City), End_Station -> Station(Name, City), Month, Daily Frequency)

2nd Solution

Bus_Station (Name, City, Cost, Construction_Company -> Construction Company(Name))

Train_Station (Name, City, Cost, Construction_Company -> Construction Company(Name))

Tram_Station (Name, City, Cost, Construction_Company -> Construction Company(Name))

Construction Company (Name, Type, Year Established)

Bus (ID, Bus_Station -> Bus_Station(Name, City), Model)

Route (Start_Station -> Tram_Station(Name, City), End_Station -> Tram_Station(Name, City), Month, Daily Frequency)

3rd Solution

Station (Name, City, Cost, Construction_Company -> Construction Company(Name))

Bus_Station (ID -> Station(Name, City))

Train_Station (ID -> Station(Name, City))

Tram_Station (ID -> Station(Name, City))

Construction Company (Name, Type, Year Established)

Bus (ID, Bus_Station -> Bus_Station(ID), Model)

Route (Start_Station -> Tram_Station(ID), End_Station -> Tram_Station(ID), Month, Daily Frequency)

2.2. Logical Relational Schema from Functional Dependencies

R_1 (A, B, C)

R_2 (D, A -> R_1(A), E)

R_3 (F, G, H)

R_4 (I, J, FK_R3 -> R_3(F, G), K, L)

Rel (FK_R2 -> R_2(D, A), FK_R3 -> R_3(F, G), O, P)

3. Functional Dependencies and Normalization

3.1. Multiple-Choice: Functional Dependencies & Normalization

1. A
2. B
3. C

3.2. Normalization

3.2.1

TrainID -> TrainCompany

OperatorID -> OperatorName

{TrainID, OperatorID, Date, EndCity} -> StartCity

3.2.2

TrainID	TrainCompany
NS12	NS
SNCB43	SNCB
DB-1214	DB
DB-5432	DB

OperatorID	OperatorName
A123	John Smith
B234	Mary Bell
AF11	Andrew Fall
SB-90	Sophie Braun

TrainID	OperatorID	StartCity	EndCity	Date
NS12	A123	Amsterdam	Delft	12/8/2021
NS12	B234	Amsterdam	Delft	3/5/2022
SNCB43	AF11	Brugge	Ghent	3/2/2022
SNCB43	AF11	Brussels	Liege	1/9/2022
SNCB43	AF11	Brugge	Liege	8/10/2022
SNCB43	AF11	Brussels	Ghent	1/9/2022
DB-1214	SB-90	Berlin	Koln	11/8/2021
DB-5432	SB-90	Frankfurt	Dusseldorf	23/5/2022
DB-5432	SB-90	Frankfurt	Dusseldorf	13/8/2022

4. SQL

1. **SELECT** s.name, **SUM**(pc.count) as passenger_count
   FROM Station s **JOIN** PassengerCount pc **ON** s.s_id=pc.s_id
   WHERE year = 2021
   GROUP BY s.name (4 points)

2. **SELECT DISTINCT** start_station.name, end_station.name
   FROM (((TrainLine tl **JOIN** Station start_station **ON** tl.start=start_station.s_id)
   JOIN Station end_station **ON** tl.end=end_station.s_id)
   JOIN Stop s **ON** tl.tl_id=s.tl_id)
   JOIN Station stop_station **ON** s.station=stop_station.s_id
   WHERE s.stop_nr = 5 **AND** stop_station.city = 'Rotterdam' (6 points)

3. **SELECT** f.name
   FROM Facilities f **JOIN** st2fac sf **ON** f.f_id=sf.f_id
   GROUP BY f.f_id, f.name
   HAVING (**MAX**(sf.capacity) - **MIN**(sf.capacity)) <=10

4. **SELECT** tl.tl_id, **COUNT DISTINCT** (s.station)
   FROM TrainLine tl **JOIN** Stop s **ON** tl.tl_id=s.tl_id
   WHERE tl.start = tl.end
   GROUP BY tl.tl_id

5. **SELECT AVG**(pc.count) as avg_passenger_count
   FROM Station s **JOIN** PassengerCount pc **ON** s.s_id=pc.s_id
   WHERE pc.month = 3 **AND** pc.year = 2022 **AND** (pc.dayOfWeek= 5 **OR** pc.dayOfWeek=6)
   AND s.name = 'Delft Station'

6. **SELECT** s.city
   FROM Station s **JOIN** st2fac sf **ON** s.s_id=sf.s_id
   GROUP BY s.city
   HAVING COUNT(DISTINCT s.s_id) >= 2 **AND COUNT**(**DISTINCT** sf.f_id) < 5