Qatar University

College of Engineering

Computer Science and Engineering Department Spring 2022 CMPS 351: Fundamentals of Database Systems

**Assignment-2**

Submission deadline: Monday, February 28th, 2022 (end-of-day, 11:59pm)

Start working on the assignment early. Avoid last minutes submissions. There will be 25% (per day) penalty for late submissions up to 2 days. No submissions will be accepted after Wednesday, March 2nd , 2022 (end-of-day, 11:59pm)

**Important Notes:**

1. Submit through Blackboard, **NO other means like email, etc., are accepted.**
2. Assignments are to be solved **individually**. You should **NOT** write-up solutions together NOR copy-and-paste solutions that are not YOURS.
3. Submit your work as instructed below ***before*** the deadline**.**

If you have any uncertainty, questions, or doubts about any of the above, ask your instructor!

**Honor Code Agreement**

**I understand that submitting work that is not my own will result in a “Zero” grade and in a disciplinary action**

**Name: Talha Abdullah**

**QUID: 201903446**

Carefully read your textbook, especially Chapter 5 and 8, then workout the following. To practice more, check the textbook exercises at the end of chapters.

1. Consider the following relations for a car-rental database application:

CUSTOMER(Customer\_no, Customer\_name, City)

RENTAL(Rental\_no, Rental\_date, Customer\_no, Time, Date, Hourly\_rate)

RENTAL\_CAR(Rental\_no, Car\_no, Driver\_no, Start\_time, End\_time, Amount\_received)

CAR(Car\_no, Year, Model, Price, Depreciation, Last\_service)

SERVICING(Garager\_no, Car\_no, Service\_date)

GARAGE(Garage\_no, Garage\_address, Owner\_name)

DRIVER(Driver\_no, Driver\_name, Driver\_address)

Here, Hourly\_rate refers to the per hour rate for a specific rental. Amount\_received is the total fare; Time and Date are the time and date when the car and driver is sent to the customer. Rental\_date is the date the rental order was placed. Assume that a car can be driven by any driver and can go to more than one garage for servicing.

1. [**1 pts**] Identify what you think should be various candidate (**primary, and unique**) keys. Write in your own words the conditions or assumptions under which each candidate key would be valid.

Ans)

**1) Customer:** Customer\_no should be the primary key as we want unique and NOT NULL values. Customer\_name and City can have multiple values, so therefore there is no need of any constraints like primary or key constraints

**2) Rental:** Rental\_no should be the primary key as we want the number to be unique and NOT NULL. Rental\_date can be similar and thus, does not require any constraints. Similarly, Time, Date and Hourly\_rate can have similar values and hence, do not require any constraints.

**3) Rental\_car**: Start\_time, End\_Time, Amount Received can be similar as two customers can have the same values for each.

**4) Car:** Car\_no can be the primary key as two cars do not have the same number plate. Model, Price, Depreciation, Last\_service can be duplicated for different cars and hence, do not require constraints.

**5) Driver:** Driver\_No can be the primary key while Driver\_name and Driver\_address can take multiple values and hence do not require constraints

**6) Garage:** Garage\_No can be the primary key as the garage is identified through its unique number. Garage\_address and Owner\_name can have similar entries and thus, do not require constraints

**7) Servicing:** No need to have any constraint for the Service\_date

1. [**2 pts**] Identify other constraints for this database? justify your answers.

Ans)

1. Customer\_name cannot be null as we need to observe customer number along with customer name

2. Driver\_name cannot be null as we reserve a driver name to a unique driver number

3. Car\_No and Garage\_No act as foreign keys in “Servicing”

4. Car\_No, Rental\_No, Driver\_No act as foreign keys in “Rental\_Car”

5. Service\_Date must not be null as Last\_Service in “Car” has to be calculated through this.

6. Time and Date must not be null in the “Rental” Table as we need to calculate Hourly\_rate

1. [**2 pts**] Populate the relations with a few sample tuples, and then give examples of insertions that violate the constraints identified in (b).
2. Table Customer: t1<0, “Talha”,”Doha”>

t2<1, “Abdullah”,” Wakrah”>

t3<0, “Muhammed”,” wakrah”>

Here t3 violates the primary constraint with the same Customer\_no value

1. Table Car: t1<132,2022,”XZ”,23245,Price-Price\*0.5,12/3/2022>

T2<232,2021,”XY”,23214,Price-Price\*0.4,12/4/2021>

Here Inserting a car with car\_no as 232 will violate primary key constraint

1. Table Rental: t1<0,12/4/2021,0,10:30,12/3/2022,500>

T2<1,12/5/2021,3,12:30,15/4/2022,5000>

Here inserting customer\_no as 3 will violate referential integrity constraint as no customer\_no is 3 in the customer table

1. Table Rental\_Car: t1<0,132,2,10:30,12:30,1000>

T2<2,153,10,12:30,6:30,5000>

Here inserting a tuple with Rental\_no as 2 or Car\_no as 153 or Driver\_no as 10 will violate referential integrity constraint

1. Table Garage: t1<100, “Doha”,”Abdullah”>

T2<100, “Wakra”,”Abdullahi”>

Here inserting another tuple with garage\_no as 100 will violate primary key constraint

1. Table Driver: t1<1, “Usama”,”Wakra”)

T2<1, NULL,”Wakra”)

Here inserting a tuple with driver\_no as 1 will violate primary key constraint and no Driver\_Name will give a no null constraint

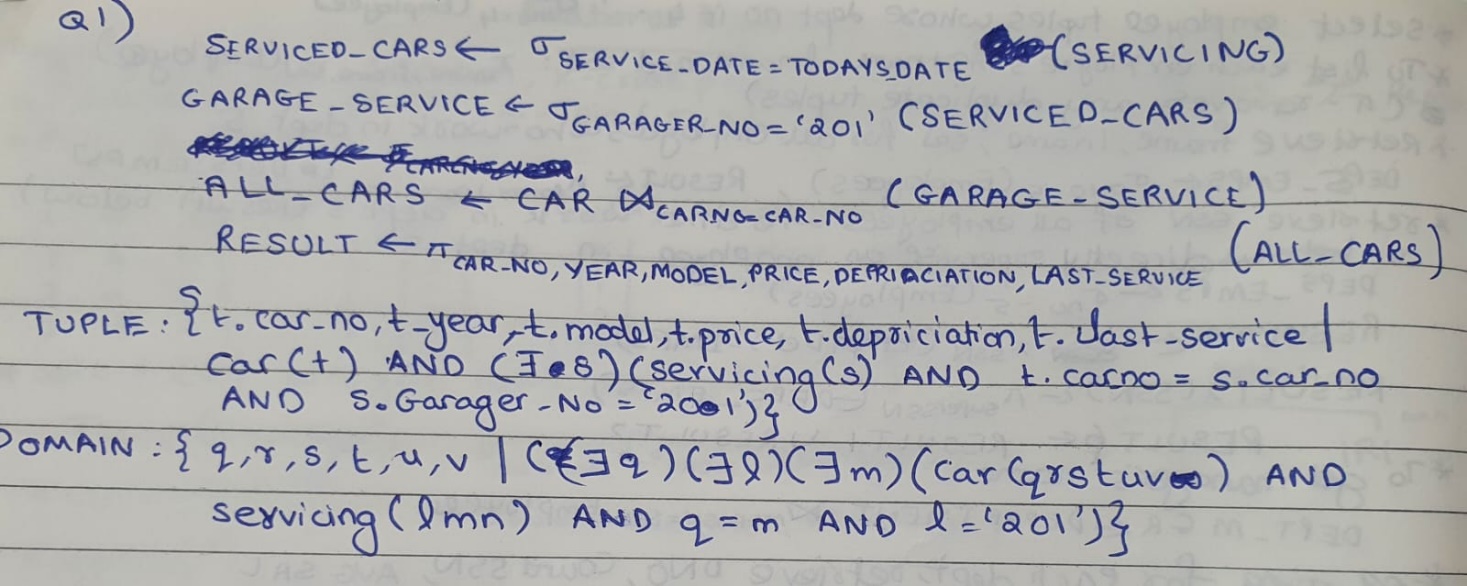
1. Table Servicing: t1<100, 132,12/3/2021>

T2< 100, 142,NULL>

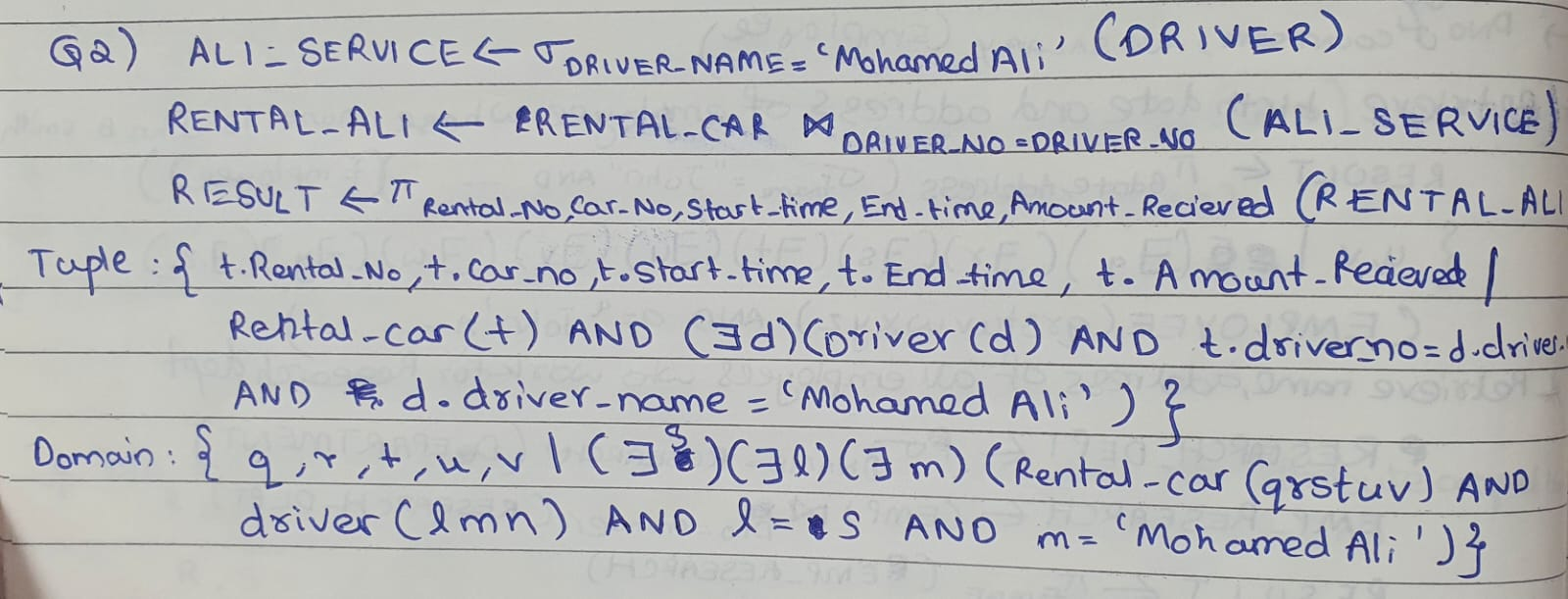
Here inserting a tuple with Car\_no as 142 will violate referential integrity constraint while leaving the Last\_service NULL will give us NO NULL Constraint

1. [**5 pts**] Write the following queries using **relational algebra** and (**tuple and domain**) **relational calculus** expressions:

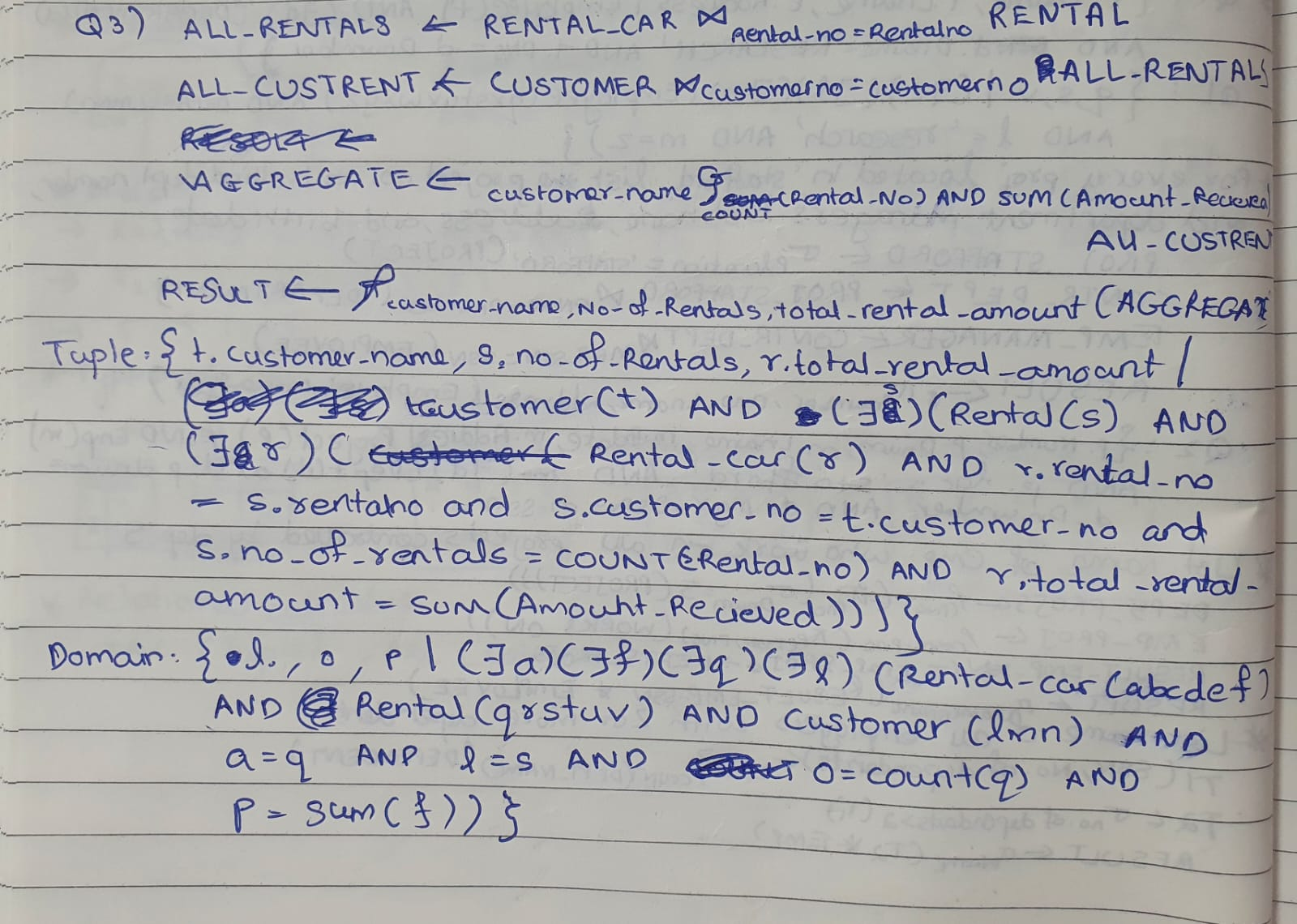
* List all information about the cars that were serviced at the garage number ‘201’ on today’s date.



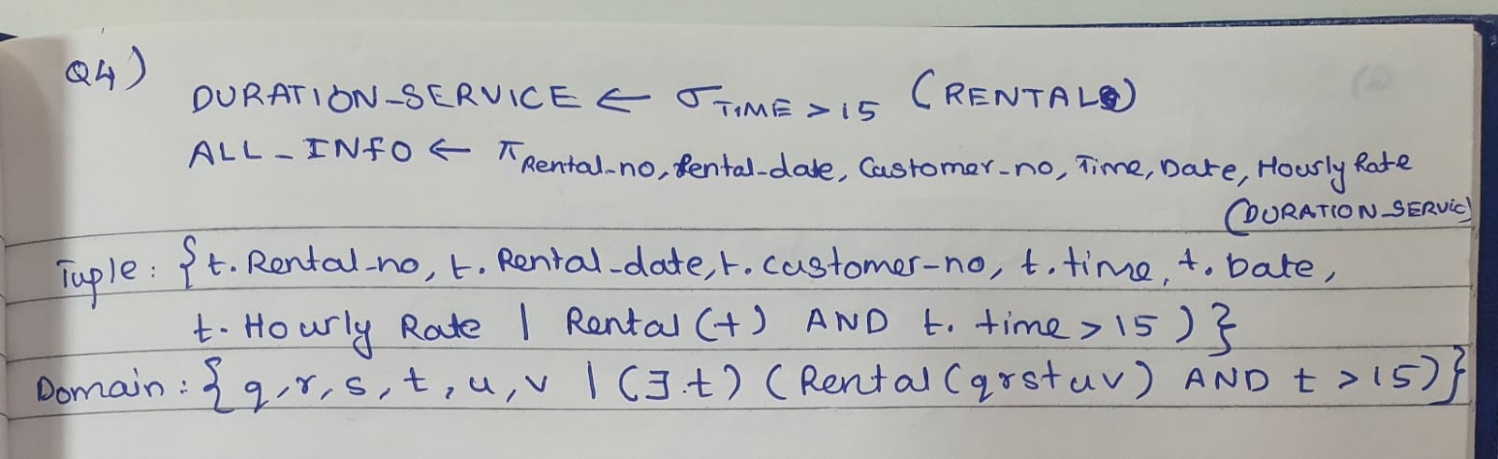
* List the RENTAL information about the rental services where the DRIVER named Mohamed Ali had provided service. Produce a listing: Rental\_no, Car\_no, Start\_time, End\_time and Amount\_received.



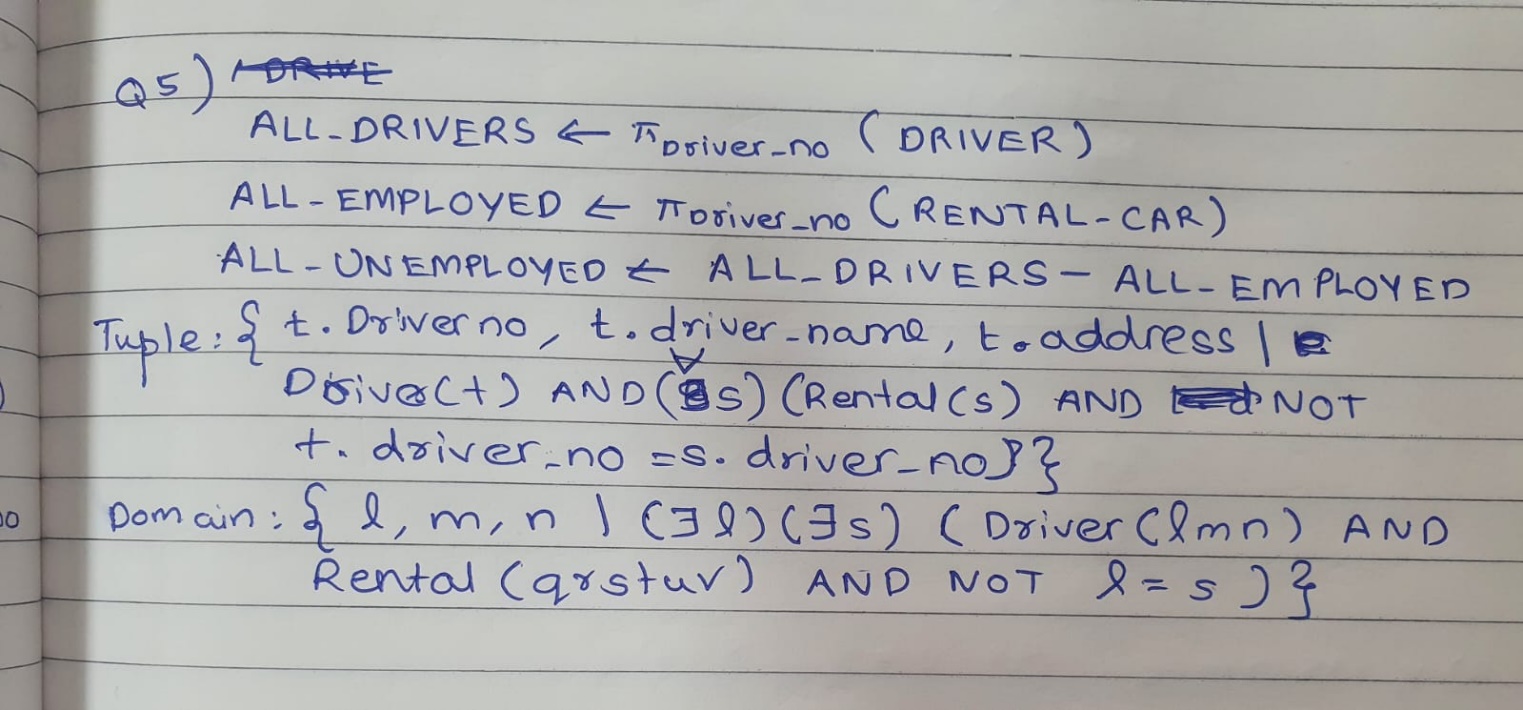
* Produce a listing Customer\_name, No\_of\_rentals, Total\_rental\_amount, where the middle column is the total number of rental services taken by the customer and the last column is the total rental amount received for that customer.



* List the rentals where the duration of service was more than 15 hours.



* List the driver number, names and addresses of all those drivers who are yet to be employed.



* **Notes:**
  + It is possible to specify the same query in multiple ways. So, give only one possible solution for each query, but with different languages.
  + You can use the symbols for the different operations as in Chapter 8 of the textbook or use the letter S for SELECT, P for PROJECT, J for JOIN, and F for FUNCTION.