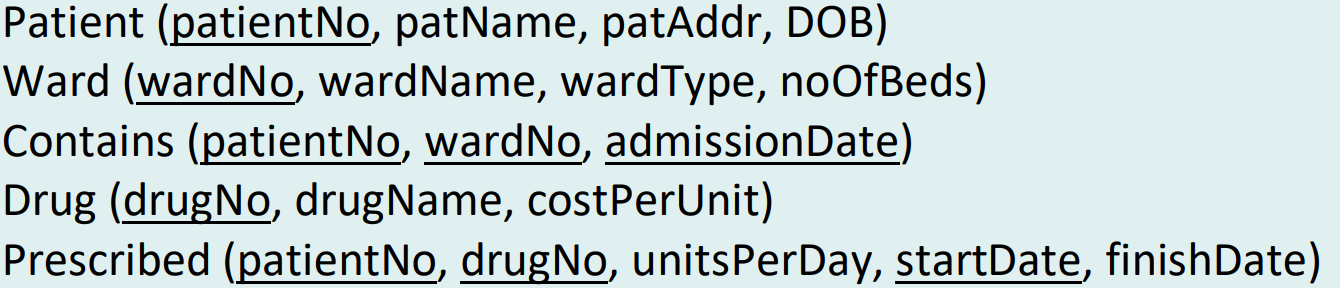
Assignment 2

By Manan & Rosmit

**Q1: More on SQL queries (4 marks)**



**i) List all the patients admitted today.**

|  |
| --- |
| SELECT p.patientNo, patName FROM Patient p, Contains c WHERE p.patientNo = c.patientNo AND admissionDate = ‘today’; |

**ii) What is the total cost of Morphine supplied to a patient called ‘John Smith’?**

|  |
| --- |
| SELECT DATEDIFF(day, finishDate, startDate) FROM Patient p, Prescribed r, Drug d WHERE p.patientNo = r.patientNo AND d.drugNo = r.drugNo AND d.drugName = 'Morphine' AND p.patName = 'John Smith'; |

**iii) What is the maximum, minimum and average number of beds in a ward? Create appropriate column headings for the results table.**

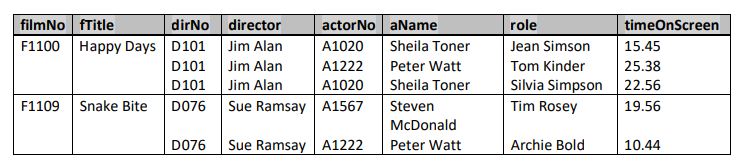
|  |
| --- |
| SELECT AVG(noOfBeds) AS “Average”, MAX(noOfBeds) AS “Maximum”, MIN(noOfBeds) AS “Minimum” FROM Ward; |

**iv) For each ward that admitted more than 10 patients today, list the ward number, ward type and number of beds in each ward.**

|  |
| --- |
| SELECT w.wardNo, wardType, noOfBeds  FROM Ward w, Patient p, Contains c  WHERE w.wardNo = c.wardNo AND p.patientNo = c.patientNo AND admissionDate = ‘today’  GROUP BY w.wardNo, wardType, noOfBeds  HAVING COUNT(\*) > 10; |

**Q2: Normalisation (4 marks)**

The table shown below displays the details of the roles played by actors/actresses in films.



**i) Describe why the table shown above is not in first normal form (1NF).**

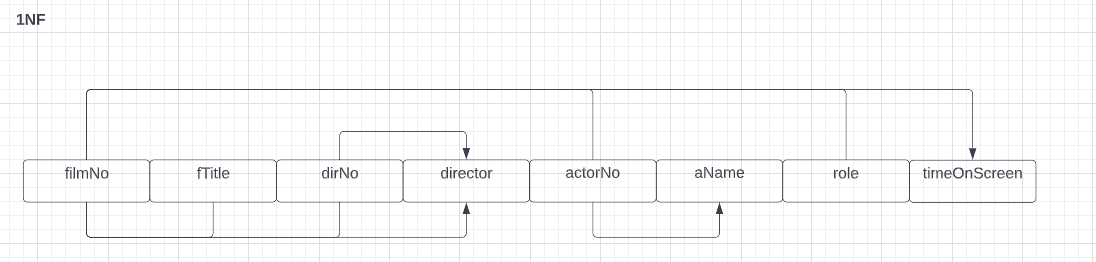
= The above table is not in 1NF because it violates the 1NF rule of having only atomic values in each field.

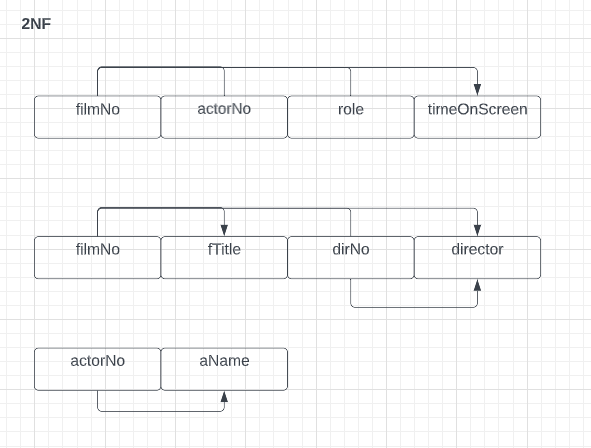
**ii) Identify the functional dependencies represented by the table shown above. State any assumptions you make about the data shown in this table, if necessary.**

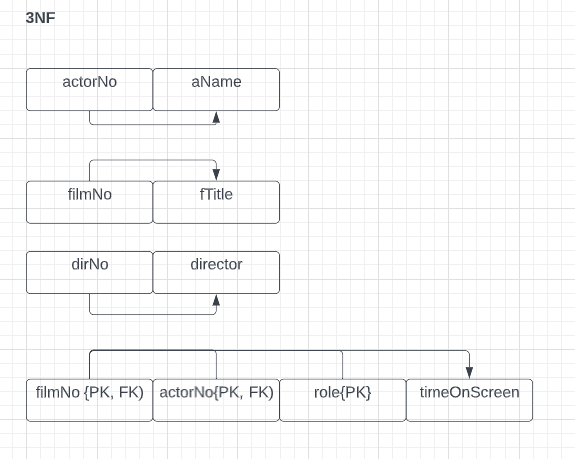
* filmNo ---> fTitle, dirNo, director
* dirNo ---> director
* actorNo ---> aName
* filmNo, actorNo, role ---> timeOnScreen

**iii) Using the functional dependencies identified in part (ii), illustrate the process of normalisation by converting the table shown above to third normal form (3NF). Identify the primary and foreign keys in your 3NF.**

(diagram in next page)

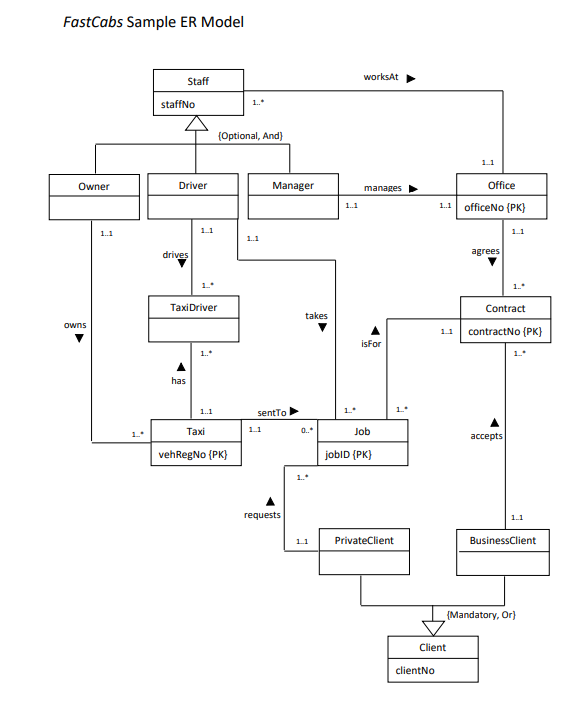
****

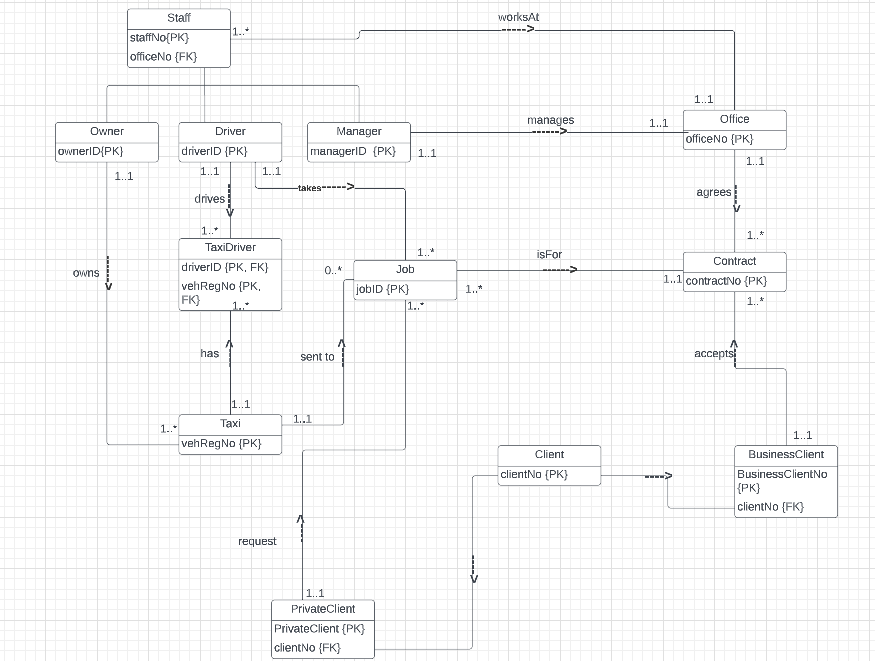
****

****

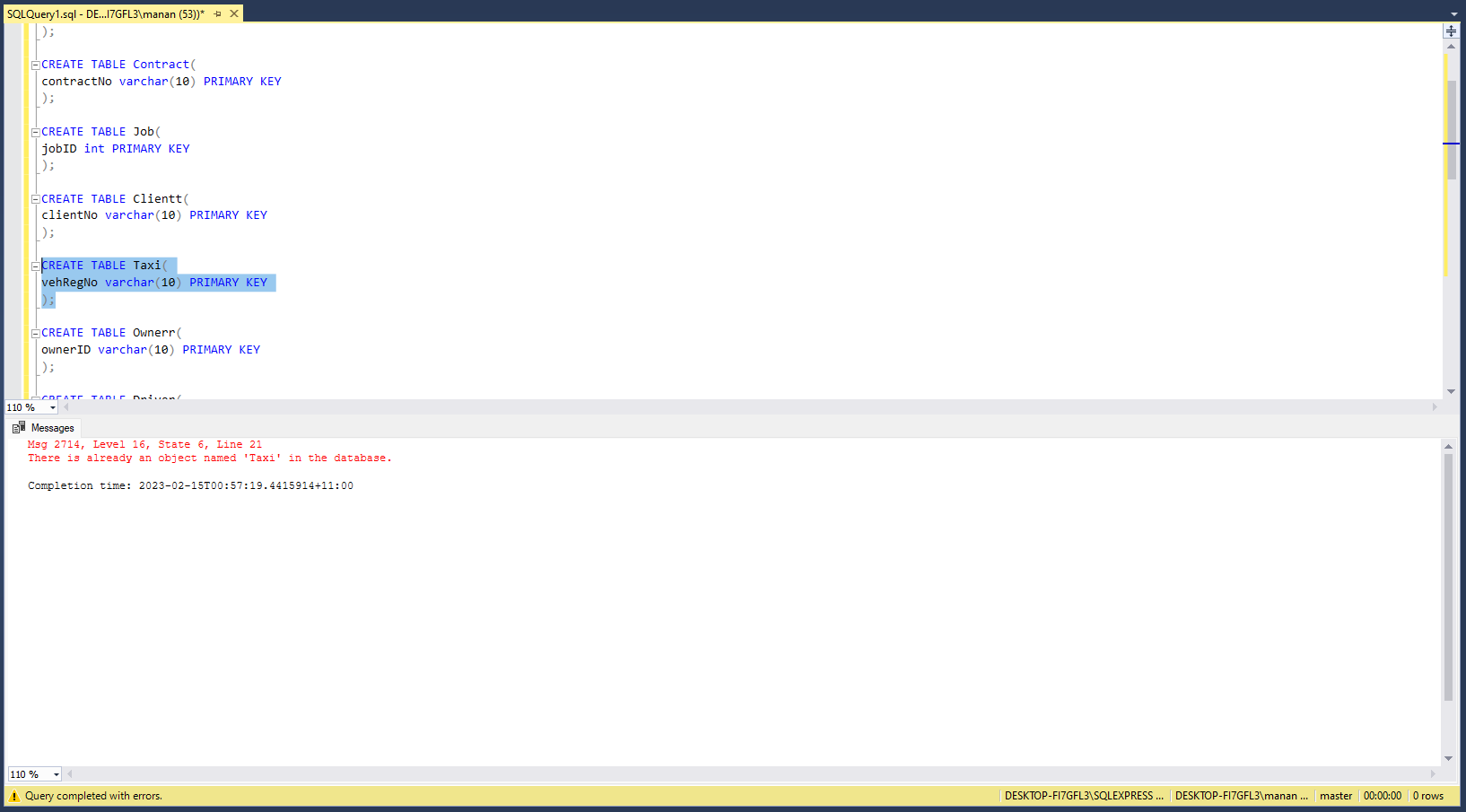
**Q3: Logical and Physical Database modelling - case study (4 marks)**

1. Given the sample ER Model for the FastCabs case study, draw the corresponding GRD, exhibiting all the primary keys and foreign keys. For simplicity, no other attributes nor multiplicity constraints are required. Please note, for the ER diagram you created in Assignment 1, the artefact of the conceptual database design, map the ER model into the relational model according to how it was designed in the ER diagram. You may however first refine or completely re-do your ER diagram if necessary, and you are allowed to make use of any part of the sample ER model provided on the next page to incorporate into your design in any way you like, without incurring any marks penalty, if you feel your original design is not in a state to be implemented later. An updated ERD is required as part of this question submission.



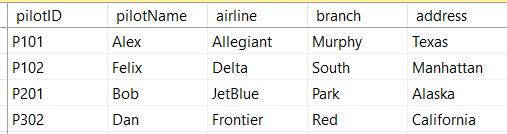
(GRD)

1. Create the database tables in SQL for all the relations in your GRD and enforce all the relevant constraints including primary and foreign keys.



**Q4. Selected Additional Exercises (3 marks)**

**P-8 (alpha)**



Here in this table if we delete the pilot - Alex from the table we're gonna lose the details about the 'Allegiant" airline as well.

**P-10 (alpha)**

Consider the following table, which stores information about sports and their respective enrollment charge

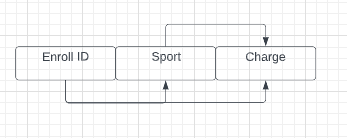
| **Enrol ID** | **Sport** | **Charge** |
| --- | --- | --- |
| 12 | Hockey | 1000 |
| 13 | Hockey | 1000 |
| 14 | Hockey | 1000 |
| 15 | Football | 2000 |
| 16 | Football | 2000 |
| 17 | Baseball | 400 |

Functional Dependencies:

Enrol ID ----> Sport, Charge

In this table, the primary key is the Enrol ID. This table is in 2NF because it has no partial dependencies. However, it is not in 3NF because there is a transitive dependency between sport and charge.

**2NF:**

****

To normalise this table to 3NF, we can create two separate tables: one for enrollment ID and their sports, and another one for sport and enrollment charge.

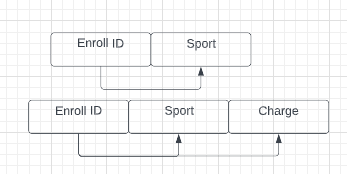
The first table, which stores information about enrollment ID and their sports:

| **Enroll ID** | **Sport** |
| --- | --- |
| 12 | Hockey |
| 13 | Hockey |
| 14 | Hockey |
| 15 | Football |
| 16 | Football |
| 17 | Baseball |

The second table, which stores the information about sport and enrollment charge:

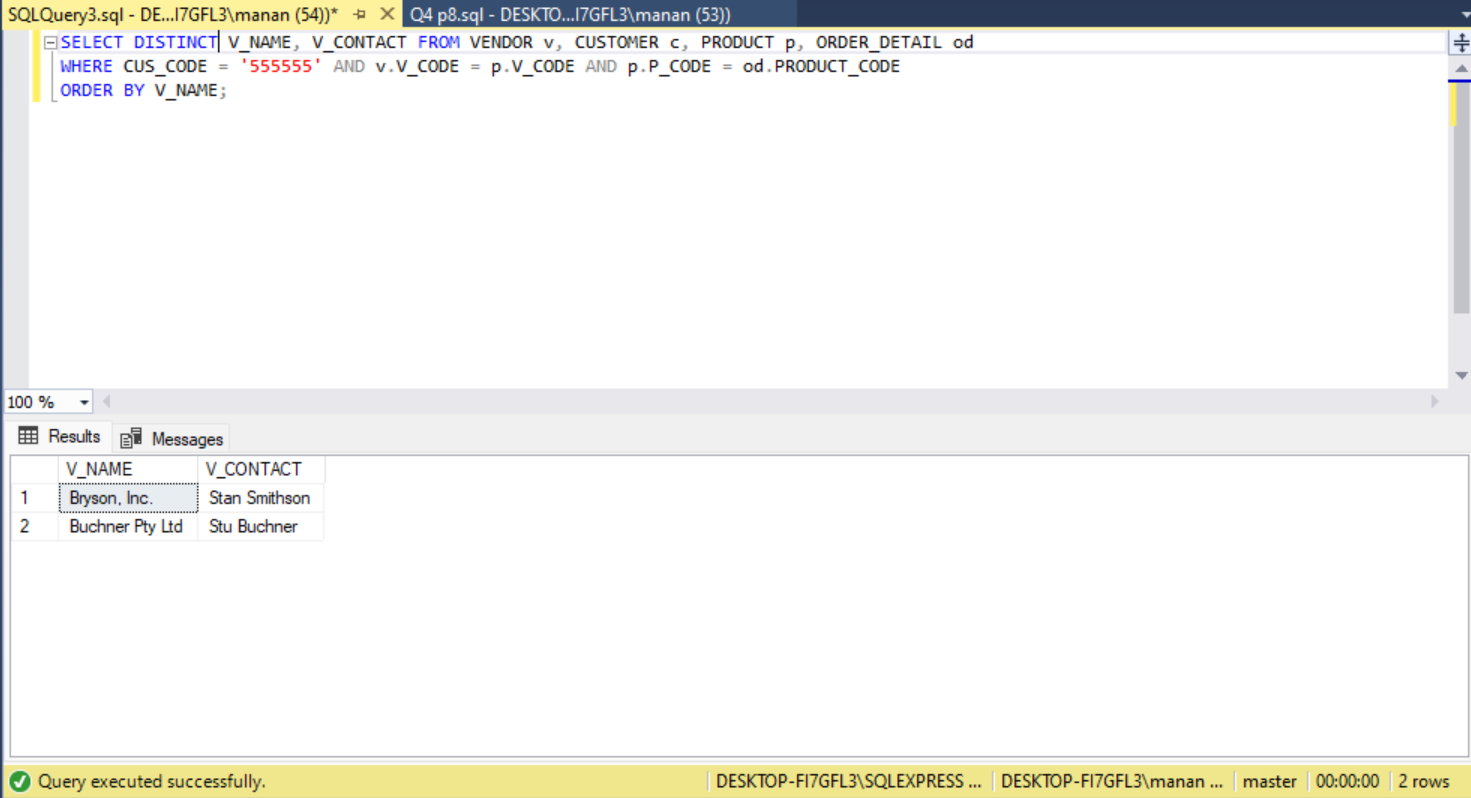
| **Sport** | **Charge** |
| --- | --- |
| Hockey | 1000 |
| Football | 2000 |
| Baseball | 400 |

**3NF:**

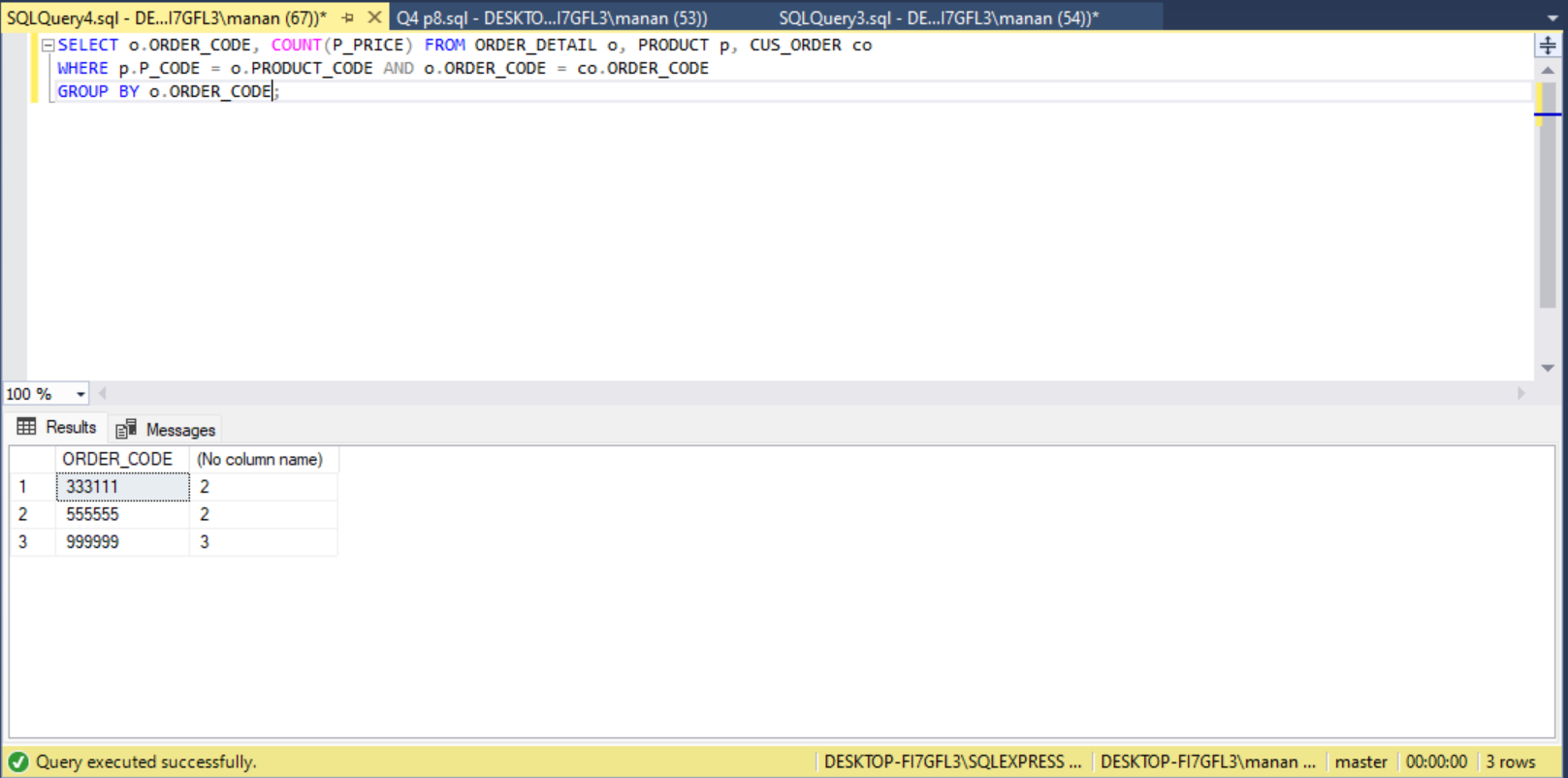
****

**P-8 (gamma)**

**b) Pick a specific customer CUS\_CODE, list the names of all the vendors who ever supplied parts to this customer. Do not repeat the vendor names in the result.**



**c)** **List all the customer order (ORDER\_CODE) and the corresponding total cost for each order.**



**P-11 (alpha)**

* **What does each letter in ACID stand for in the context of database transactions? Describe a *concrete example* (i.e. a scenario) to illustrate the property the letter "I" refers to.**

ACID stands for:

* Atomicity
* Consistency
* Isolation
* Durability

**A** - Atomicity: It means either the entire transactions are committed or none of them is committed.

**C** - Consistency: Data integrity constraints must not be violated so that the database is consistent before and after the transaction.

**I** - Isolation: Isolation ensures the concurrent occurrence of multiple transactions without any inconsistency of the database state. Any transaction will not be impacted by another transaction happening on the same database.Changes occurring in a particular transaction will not be visible to any other transaction until that particular change in that transaction is committed.

Example:

Let A = 200, B = 300

Consider two transactions t1 and t2.

| **t1** | **t2** |
| --- | --- |
| Read(A)  A = A + 50  Write(A)  Read(B)  B = B - 10  Write | Read(A)  Read(B)  C = A + B  Write(C) |

Suppose t1 has been executed till Read (B) and then t2 starts. As a result, interleaving of operations takes place due to which t2 reads the correct value of A but the incorrect value of B and sum calculated by

t2’: (A+B = 250 + 300 =550)

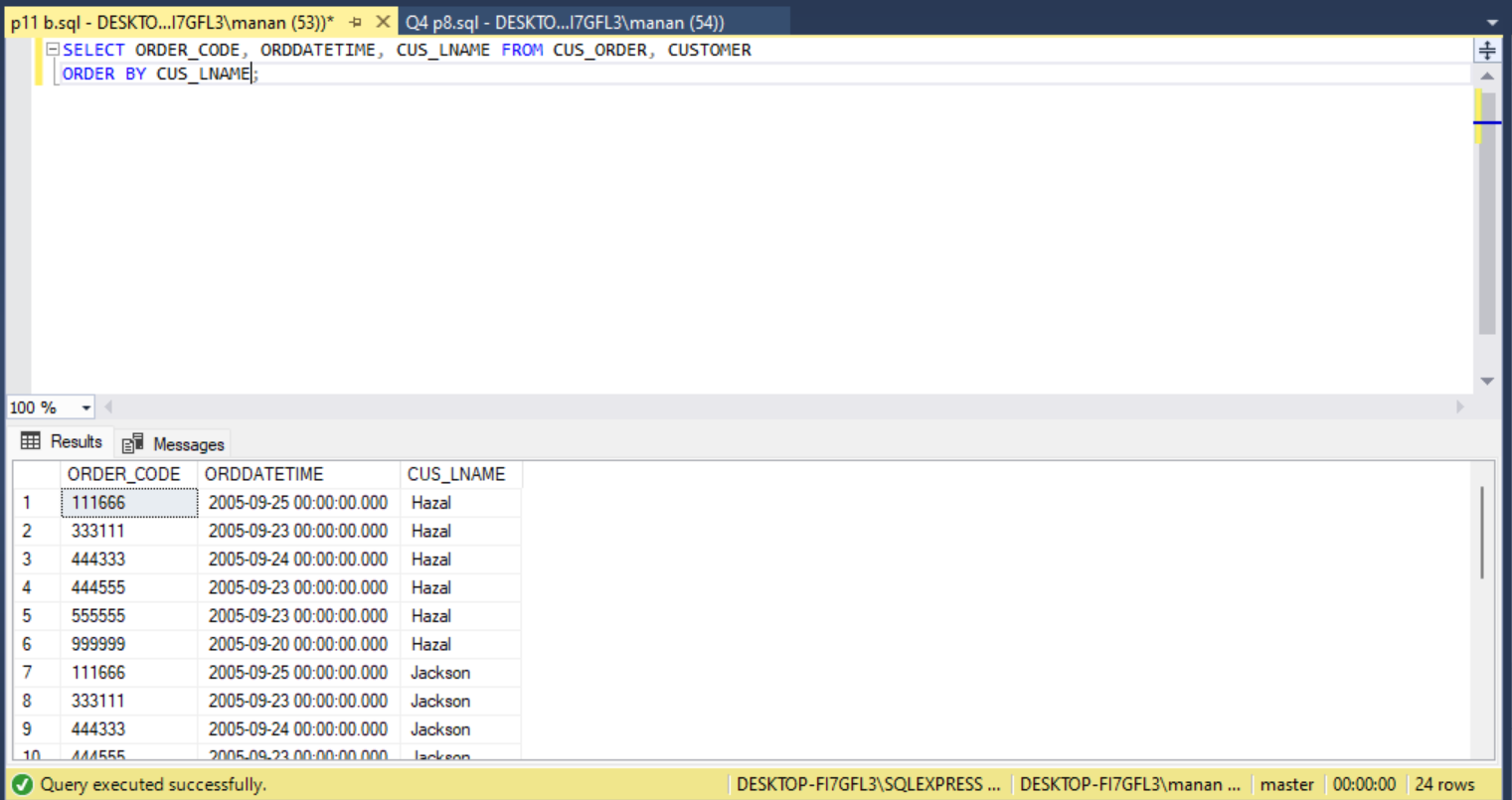
is thus not consistent with the sum at end of transaction:

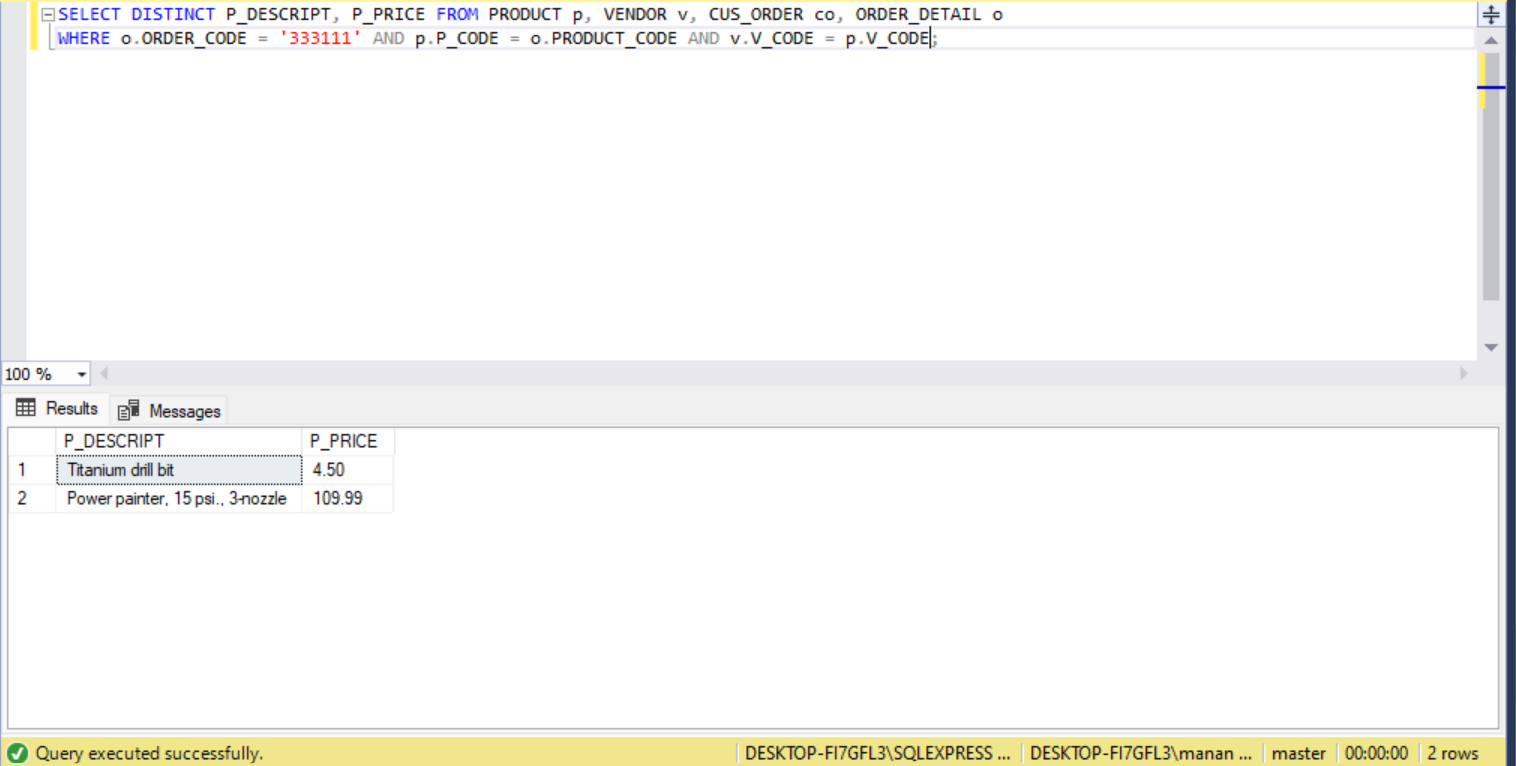
t1: (A+B = 200 + 250 = 450).

This results in database inconsistency.

**D** - Durability: It ensures that the changes made through the successfully committed transaction will be saved permanently in the database, even if there is the case of system failure.

**P-11 (beta)**

**b) List order number (ORDER\_CODE), order date (ORDDATETIME) and the customer's last name (CUS\_LNAME) for all orders, including the orders which contain no customer names.**

**c) For a given customer order number, say '333111', list all the order product description (P\_DESCRIPT), price (P\_PRICE) and the corresponding vendor name (V\_NAME) if available.**

**Work Contribution:**

Manan has worked on the SQL scripts questions while Roshmit worked on Normalisation questions. Both Manan and Roshmit worked on the GRD of Q3 together and contributed 50/50.