Mock Exam – Database Fundamental and Applications

General remarks. In this exam you are **NOT** allowed to use a

- calculator
- book
- dictionary

You have two hours for this exam.

PART I: Concepts [30 points]

Question 1. [3 points] Describe how a theta join works in terms of relations, attributes, and tuples. Now, given relations $R(A, B)$ and $S(B, C)$ also describe how a theta join on the condition $R.B = S.B$ differs from a natural join.

Question 2. [3 points] In the relational model of data, relations are conceptualised as sets of tuples. SQL, however, treats relations as bags of tuples instead. Explain the differences between sets of tuples and bags of tuples, and give two advantages of the bag-based approach.

Question 3. [3 points] Consider a relation $R(A, B, C, D, E)$ that is decomposed into relations $R_1(A, B, C)$ and $R_1(C, D, E)$. What does it mean for the natural join of R_1 and R_2 not to be lossless?

A. There may be tuples t in R that are not found in $R_1 \bowtie R_2$.

B. There may be tuples t in $R_1 \bowtie R_2$ that are not found in R.

C. Both A and B.

D. There must be tuples t in $R_1 \bowtie R_2$ that are not found in R and, for some instances of R, there may be tuples t in R that are not found in $R_1 \bowtie R_2$.

Question 4. [3 points] Describe the difference between relations and relationships.

Question 5. [3 points] Explain the difference between SQL tables, virtual views, and materialised views.

Question 6. [3 points] When is a relation said to be in Boyce-Codd Normal Form (BCNF)? What is the use of decomposing a relation into this form? Explain what the downsides are, if any, to decomposing a relation into BCNF.

Question 8. **[3 points]** Explain what a dirty read is, and also give both an advantage and disadvantage of allowing dirty reads. You may use practical examples in your explanation.

Question 9. **[3 points]** Proper transactions should meet the so-called ACID properties. Explain what these properties are.

Question 10. **[3 points]** Explain how indices can help query your tables more efficiently, and also explain what trade-offs are involved when deciding on which and how many (sets of) attributes to set as indices.

PART II: Applications of theory [30 points]

Question 11. **[15 points]** Consider relation $R(A, B, C, D, E)$ with the following functional dependencies (FDs): $A \to D$, $BC \to E$, $E \to A$.

i) Find all non-trivial, implied FDs with one attribute on the right-hand side.

ii) List all keys.

iii) Indicate which FDs violate Boyce-Codd Normal Form.

iv) Let's decompose R into $R_1(B, C, E)$ and $R_2(A, D, E)$. Project all given and derived FDs onto R_1 and R_2.

v) Are the decomposed relations in BCNF?

vi) For the FDs projected onto R_1: find the minimal basis. Do the same for the FDs projected onto R_2.

vii) Based on these minimal bases for the FDs of R_1 and R_2, what set of FDs do they imply for $R_1 \bowtie R_2$? Is this set equivalent to the set of FDs that holds for R?

viii) Use the chase test to assert whether the join of R_1 and R_2 is lossless.

ix) Would there be any advantage to decomposing this relation into third normal form (3NF) using the synthesis algorithm? Explain why or why not.

Question 12. **[15 points]** The director of a theatre asks you to set up a database system for the sales of tickets to customers for different performances of a show. The director provides you with the following information:

- Each show has a name and producer.
- For each show there are one or more performances (unless the show is cancelled altogether).
- In each performance exactly one show is performed.

- Each performance has a date, time, and hall in which it takes place.
- Each performance has a crew consisting of one or more crew members.
- Each member of the crew has a name, birth date, and role.
- Tickets are sold for each performance.
- Each ticket constitutes a reservation of one seat for one performance.
- Each ticket has a price and a seat number.
- Each ticket can be bought by at most one customer.
- Each customer can hold multiple tickets. Each customer has a name and address in the system.

Using only this information:

i) Draw an appropriate entity-relationship (ER) diagram for this database.

ii) Are there any weak entity sets in your diagram? If so, explain why they are weak entity sets.

iii) What are the referential-integrity constraints?

iv) Convert the ER-diagram to a set of relations, in accordance with the rules you have learnt in Chapter 4.

v) Setting aside all entity sets (ESs) other than Shows: draw an ER-diagram focussing exclusively on Shows, which allows you to distinguish between a ballet, a musical, and a regular play, where

   - a ballet has a fixed crew of one or more ballet dancers as well as an orchestra consisting of multiple musicians,
   - a musical has a fixed crew of one or more actors as well as an orchestra consisting of multiple musicians, and
   - a regular play has a fixed crew of one or more actors.

PART III: SQL queries [30 points]

We have data from an airline company on three relations, `Flights`, `Seats`, and `Bookings`, shown in Tables 1, 2, and 3 respectively.

flightNo	date	plane	time	dest
1031	2018-12-01	908	09:12	LCY
1032	2018-12-01	908	12:29	RTM
0134	2018-12-02	671	10:09	VIE
0135	2018-12-02	671	15:43	AMS

Table 1. Relation `Flights`

plane	row	number	class
671	1	A	business
671	1	B	business
671	1	E	business
671	1	F	business
671	2	A	business
671	2	B	business
671	2	E	business
671	2	F	business
671	3	A	economy
671	3	B	economy
671	3	E	economy
671	3	F	economy
671	4	A	economy
671	4	B	economy
671	4	E	economy
671	4	F	economy
908	1	A	business
908	1	B	business
908	1	E	business
908	1	F	business
908	2	A	business
908	2	B	business
908	2	E	business
908	2	F	business
908	3	A	economy
908	3	B	economy
908	3	E	economy
908	3	F	economy
908	4	A	economy
908	4	B	economy
908	4	E	economy
908	4	F	economy

Table 2. Relation `Seats`

flightNo	date	seatRow	seatNo	price	customerID
1031	2018-12-01	1	A	800	1209495
1031	2018-12-01	3	A	120	1689312
1031	2018-12-01	3	B	130	3429123
1031	2018-12-01	3	E	110	4292135
1031	2018-12-01	3	F	130	9329401
1031	2018-12-01	4	A	180	3949123
1031	2018-12-01	4	E	140	3126902
1032	2018-12-01	1	B	650	4783013
1032	2018-12-01	2	F	750	4539231
1032	2018-12-01	3	A	110	1263912
1032	2018-12-01	3	B	90	2131230
1032	2018-12-01	3	E	100	4502343
1032	2018-12-01	4	A	120	1839694
1032	2018-12-01	4	E	100	8924922
1032	2018-12-01	4	F	130	7255242
0134	2018-12-02	1	A	1500	7324723
0134	2018-12-02	1	E	1700	2139067
0134	2018-12-02	2	F	1100	6589123
0134	2018-12-02	3	E	210	4589212
0134	2018-12-02	4	B	190	9648324
0135	2018-12-02	1	B	1300	5839149
0135	2018-12-02	2	E	1200	8573271
0135	2018-12-02	3	A	180	5182542
0135	2018-12-02	3	B	180	6581298
0135	2018-12-02	4	A	150	1231205
0135	2018-12-02	4	B	160	8132984
0135	2018-12-02	4	F	170	3210935

Table 3. Relation `Bookings`

Question 13. [**10 points**] Write a SQL query to compute the total amount money spent on bookings for each flight and each class. Also provide the result of this query.

Question 14. [**10 points**] Write a SQL query to compute the number of seats available on each flight, in each class. Also provide the result of this query.

Question 15. [**10 points**] Write a SQL query to show which rows on which flights are still completely empty (i.e. for which there is no single booking). Note that by flight I mean the combination of flightNo and date. Also provide the result of this query.