Chapter 14 Review Questions

1) Why should NULLs in a relation be avoided as much as possible? Discuss the problem of spurious tuples and how we may prevent it.
    a) NULL values represent a lack of data which can cause defects in certain operations. NULL values also point to the possibility that the attribute should be another relation because it isn't always applicable, known, or available. NULLS also waste storage space in the database. Spurious tuples occur when bad relational design causes join operations to result in too many tuples. Spurious tuples can be avoided by only using the key attributes of entities in relationships and to avoid different kinds of attributes from having the same name. This forces tables to natural-join only based on the key attributes. Fixing a database to not have spurious tuples occur usually requires splitting tables into more basic relations.
2) Define 1st, 2nd, and 3rd normal forms when only primary keys are considered
    a) 1NF is considered the basic definition of a relation and can have any amount of primary keys. There is nothing constraining the keys other than attributes being dependent on them. "All attributes depend on the key"
    b) 2NF requires that removing any of the primary keys eliminates the functional dependency. "All attributes depend on the whole key"
    c) 3NF requires that the attributes are dependent on only the primary key(s). By normalizing to 3NF, a transitive property of relations arises. "All attributes depend only on they key"
3) What undesirable dependencies are avoided when a relation is in 2NF?
    a) 2NF avoids partial dependencies of non-prime attributes. Any attribute that only depends on one of multiple primary keys is undesirable. This causes the dependency to be a full functional dependency
4) What undesirable dependencies are avoided when a relation is in 3NF?
    a) 3NF avoids non-prime attributes from being transitively dependent on the primary key. Any attribute that is dependent on another non-prime attribute is undesirable and avoided by splitting up the table.

5) List all functional dependencies of the following relation:
   a) TUPLE# -> any combination of other attributes
   b) B -> C
   c) (A,B) -> C
   d) (A,C) -> B

A	B	C	TUPLE#
10	b1	c1	1
10	b2	c2	2
11	b4	c1	3
12	b3	c4	4
13	b1	c1	5
14	b3	c4	6

6) Consider the following relation: Based on the given primary key, is this relation in 1NF, 2NF, or 3NF? Why or why not? How would you successfully normalize it completely?
   a) This relation is not in any normal form because a car sale can have multiple salespeople, so it has a multivalued attribute.
   b) To get the relation into 1NF you would split the Salesperson# into multiple rows. To get the relation into 2NF you would split the table into CAR_SALE(Car#, Date_sold, Salesperson#, Discount_amt) and COMMISSION(Salesperson#, Commission%). TO reach 3NF you would split CAR_SALE into CAR_SALE(Car#, Date_sold, Salesperson#) and DISCOUNT(Date_sold, Discount_amt).

CAR_SALE(Car#, Date_sold, Salesperson#, Commission%, Discount_amt)
Assume that a car may be sold by multiple salespeople, and hence {Car#, Salesperson#} is the primary key. Additional dependencies are

Date_sold → Discount_amt and Salesperson# → Commission%