Quiz: Database Design

1) You have a field in Table A that you’d like to include in Table B. You want to make sure you don’t duplicate the data in Table B, and that the data stays consistent across the two tables. What should you do?

a) Unfortunately, you would need to duplicate the data  
\*b) Include the field as a foreign key in Table B  
c) Include the field as a primary key in Table B  
d) Create and index on the field in Table A

Reasoning: you can enforce the field relationship and avoid redundancy but referencing the field as a foreign key  
  
2) You are tasked with designing a database for a client. Which of the following questions is essential to ask before building the database? Select all that apply.

\*a) What should the system do?  
\*b) What is the required response time?  
\*c) Do you prefer zero-based indexing or one-based indexing?  
\*d) What are the data constraints?  
\*e) Is there documentation on the existing system?

3) You are about to build a database. One of the employees – without any database experience – mentions that a graph database is absolutely required. What should you do?

a) Politely say “Thank you” and ignore his suggestion  
b) Do as the customer says (It’s his money)  
\*c) Ask him to explain his reasoning  
d) Say thanks, walk away, and go study the problem to see if that kind of database makes sense

4) True or False: An Entity Relationship Diagram contains the full details needed to build a database.  
  
a) True  
\*b) False  
  
Reasoning: An ER Diagram shows the objects and relationships, but it’s missing things like data type

5) Which of these examples should not be in a relational database? Select all that apply.

a) You have highly structured insurance claim data that has many relationships between the fields  
\*b) You scraped the web, producing millions of unstructured documents that need storage  
\*c) A social networking firm tracks hundreds of millions of people and their relationships  
d) A winery wishes to track a hundreds of wines that they’ve produced over the years, including information about varietals and soil conditions

Reasoning: b) should use a document-oriented database, c) should use a graph database

6) You are designing a database and understanding key business rules. Some business rules are simple, while others are quite complex. These business rules:

a) all need to be included in the database  
b) can be handled both in the database and in code. The simpler rules should be coded, while the more complex rules should be in the database.

\*c) can be handled both in the database and in code. The simpler rules should be included in the database, while the more complex rules should be coded.

d) should be included in the database as best as possible. Remaining, complex rules can be included in the database in the next release.

7) Database normalization (select all that apply):

a) should always be done to fifth normal form (5NF)

\*b) is a tradeoff, as it can make the database more useful, but it often increases the need for joining tables  
c) means centering and scaling all quantitative fields

\*d) results in tables that contain one kind of data (for example, there might be a table about movies and another about viewers; there would NOT be a table contain both movie and viewer information)

Reasoning:   
a) data should generally be done to 3NF; 5NF is usually excessive  
c) this should be done to data when building certain models

8) Which of these is a database or includes a database as part of the system? Select all that apply

\*a) a cat’s brain

\*b) an application where you can place orders, change them, delete them, and retrieve them

\*c) an Excel spreadsheet

d) a movie

Reasoning: A database must support Create, Retrieval, Update, and Delete operations.   
d) a movie does not support these

9) Your database has a table that holds stock tickers with their associated prices on different closing dates. Which of these would work as a primary key?

a) ticker  
\*b) ticker, closing date

c) ticker, price

d) ticker, closing date, price

Reasoning: the primary key uses minimal fields that make each row unique  
  
10) By carefully planning out the database before buildout, you can:  
  
a) avoid the need to ever change the database further  
\*b) minimize the number of schema changes later  
\*c) surface all reasonable use cases early, making the database more useful later   
\*d) better satisfy the important stakeholders