Databases Core Concepts & Database Design Interview - Explanations

Section 2, Lecture 3

1. What is a Relational Database?

Relational Databases are the most common database systems used today. Some examples are: SQL Server, Oracle Database, IBM Informix, MySQL and many others..

The relational database management systems (RDMS) feature very good performance for managing data, they allow multiple users (even thousands!) to work with the data at the same time, and are providing advanced security for accessing the data. Relational databases store data in columns and rows, which in turn make up tables. These tables are linked to each other using relationships. Thus the name relational. A set of tables makes up a schema. A number of schemas create a database. Many databases can be created on a single server.

2.A DBMS stands for

Database Management System, asoftware system used to maintain databases.

A database management system (DBMS) is a software package designed to define, manipulate, retrieve and manage data in a database. A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data. A DBMS relieves users of framing programs for data maintenance. Fourth-generation query languages, such as SQL, are used along with the DBMS package to interact with a database.

3.Which of the following are examples of a Database?

Except the Box of Photos (which could be scanned and stored on a file system - no need for a database) every item mentioned above could be modeled and its data persisted in a Relational Database.

4. Which of the following are benefits of a well-designed database?

•It prevents duplicate data

•The database structure is easy to modify and maintain. (It is easy to expand in the future)

•The data is easy to modify.

•Information is easy to retrieve.

•End-User applications are easy to develop and build.

5.An Entity is a ________

Entities are usually recognizable concepts, either concrete or abstract, such as person, places, things, or events which have relevance to the database. Some specific examples of entities are Employee, Student, Lecturer. An entity is analogous to a table in the relational model.

6.What are attributes?

The columns are called attributes, which are the describing characteristics of each tuple (row). Say customers, flights, or products. Think of describing a product. You want to include itsname,price, weight, color, etc. These would be important characteristics that help to identify the product.

7. What are other names used for columns?

In a relational database, a column is a set of data values of a particular simple type, one value for each row of the database.A column could contain text values, numbers, or even pointers to files in the Operating System.Some relational database systems allow columns to contain more complex data types, for example whole documents, images or even video clips.**A column can also be called an attribute or field.**

8.Which of the following statements are true?

(question related to what columns,rows and tables are**)**

Answer from above for defining a column.

In the context of a relational database, a row—also called a record or tuple—represents a single, implicitly structured data item in a table. In simple terms, a database table can be thought of as consisting of rows and columns or fields.

9. A tableconsists of rows and column.

A collection of closely related columns. A table consists of rows each of which shares the same columns but vary in the column values.

10.What are the table rows also called?

In the context of a relational database, a row—also called**a record or tuple**—represents a single, implicitly structured data item in a table. In simple terms, a database table can be thought of as consisting of rows and columns or fields.Each row in a table represents a set of related data, and every row in the table has the same structure.

For example, in a table that represents companies, each row would represent a single company. Columns might represent things like company name, company street address, whether the company is publicly held, its VAT number, etc.. In a table that represents the association of employees with departments, each row would associate one employee with one department.

11.What is the name of a field or set of fields used to uniquely identify a record in a table? (referringto primary key)

A**primary key**is a special relational database table column (or combination of columns) designated to uniquely identify all table records. A primary key's main features are: It must contain a unique value for each row of data. It cannot contain null values.

12.Which of the following is a valid key field

A. Yes, a car registration number is specially designed to be globally unique - if you know the registration number, you can trace any car.

B. No.It's possible to have different towns with the same street names.

C. No.There is no reason why two or more people shouldn't share names and birthdays.

D. No, books do have a unique key field though it's called the ISBN.

13.What are the three main data types used when designing a database?

In most databasesthere are three main data types:**text, number, and date.**

Bellow are ALLMySQLData Types. There is no need to memorize their details; just review them once before an interview in case someone asks:"What are 3 data types used to store dates?" ..or something among these lines.

Text data types:

Data type	**Description**
CHAR(size)	Holds a fixed length string (can contain letters, numbers, and special characters). The fixed size is specified in parenthesis. Can store up to 255 characters
VARCHAR(size)	Holds a variable length string (can contain letters, numbers, and special characters). The maximum size is specified in parenthesis. Can store up to 255 characters.**Note:**If you put a greater value than 255 it will be converted to a TEXT type
TINYTEXT	Holds a string with a maximum length of 255 characters
TEXT	Holds a string with a maximum length of 65,535 characters
BLOB	For BLOBs (Binary Large OBjects). Holds up to 65,535 bytes of data
MEDIUMTEXT	Holds a string with a maximum length of 16,777,215 characters
MEDIUMBLOB	For BLOBs (Binary Large OBjects). Holds up to 16,777,215 bytes of data
LONGTEXT	Holds a string with a maximum length of 4,294,967,295 characters
LONGBLOB	For BLOBs (Binary Large OBjects). Holds up to 4,294,967,295 bytes of data
ENUM(x,y,z,etc.)	Let you enter a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. **Note:**The values are sorted in the order you enter them. You enter the possible values in this format: ENUM('X','Y','Z')
SET	Similar to ENUM except that SET may contain up to 64 list items and can store more than one choice

Number data types:

Data type	**Description**
TINYINT(size)	-128 to 127 normal. 0 to 255 UNSIGNED*. The maximum number of digits may be specified in parenthesis
SMALLINT(size)	-32768 to 32767 normal. 0 to 65535 UNSIGNED*. The maximum number of digits may be specified in parenthesis
MEDIUMINT(size)	-8388608 to 8388607 normal. 0 to 16777215 UNSIGNED*. The maximum number of digits may be specified in parenthesis
INT(size)	-2147483648 to 2147483647 normal. 0 to 4294967295 UNSIGNED*. The maximum number of digits may be specified in parenthesis
BIGINT(size)	-9223372036854775808 to 9223372036854775807 normal. 0 to 18446744073709551615 UNSIGNED*. The maximum number of digits may be specified in parenthesis
FLOAT(size,d)	A small number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter
DOUBLE(size,d)	A large number with a floating decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter

DECIMAL(size, d)	A DOUBLE stored as a string , allowing for a fixed decimal point. The maximum number of digits may be specified in the size parameter. The maximum number of digits to the right of the decimal point is specified in the d parameter

*The integer types have an extra option called UNSIGNED. Normally, the integer goes from an negative to positive value. Adding the UNSIGNED attribute will move that range up so it starts at zero instead of a negative number.

Date data types:

Data type	**Description**
DATE()	A date. Format: YYYY-MM-DD **Note:**The supported range is from '1000-01-01' to '9999-12-31'
DATETIME()	*A date and time combination. Format: YYYY-MM-DD HH:MI:SS **Note:**The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'
TIMESTAMP()	*A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD HH:MI:SS **Note:**The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC
TIME()	A time. Format: HH:MI:SS **Note:**The supported range is from '-838:59:59' to '838:59:59'
YEAR()	A year in two-digit or four-digit format. **Note:**Values allowed in four-digit format: 1901 to 2155. Values allowed in two-digit format: 70 to 69, representing years from 1970 to 2069

14. In Relational Database terminology which of the following is true:

•A row is alsocalledtupleorrecord

•A column is also calledfieldorattribute

•A table is called arelation

15.What are some properties of a Relational Databases?

In relational databases, each column should have a unique name. Sequence of rows and columns in relational databases are insignificant. All values are atomic and each row is unique.

16.What is a Database Schema?

The database schema of a database system is its structure described in a formal language supported by the database management system (DBMS). The term "schema" refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). In a relational database, the schema defines the tables, fields, relationships, views, indexes, packages, procedures, functions, queues, triggers, types, sequences, materialized views, synonyms, database links, directories, XML schemas, and other elements.

17.If there is more than one key for relation schema in DBMS then each key in relation schema is classified as: candidate key.

(referringto primary keys, candidate keys, superkey, prime and non-prime attributes)

A candidate key is a column, or set of columns, in a table that can uniquely identify any database record without referring to any other data. Each table may have one or more candidate keys, but one candidate key is unique, and it is called the primary key.

18.If attribute of relation schema R is not a member of some candidate key then this type of attribute is classified as:non primeattribute.

(referringto primary keys, candidate keys, superkey, prime and non-prime attributes)

The constituent attributes are called prime attributes. Conversely, an attribute that does not occur in ANY candidate key is called a non-prime attribute. Since a relation contains no duplicate tuples, the set of all its attributes is a superkey if `NULL` values are not used.

19.What is Database Normalization?

Database normalization, or simply normalization, is the process of restructuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. It was first proposed by Edgar F. Codd as an integral part of his relational model.

Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure that their dependencies are properly enforced by database integrity constraints. It is accomplished by applying some formal rules either by a process of synthesis (creating a new database design) or decomposition (improving an existing database design).

20.What are the stages of Database Normalization?

Normalisation follows a staged process that obeys a set of rules. The steps of normalisation are:

Step 1: Transform the unnormalised data into first normal form (1NF)

Step 2: Transform data in first normal form (1NF) into second normal form (2NF)

Step 3: Transform data in second normal form (2NF) into third normal form (3NF)

Occasionally, yet rarely used in practice,the data may still be subject to anomalies in third normal form. In this case, we may have to perform further transformations.

Transform third normal form to Boyce-Codd normal form (BCNF)
Transform Boyce-Codd normal form to fourth normal form (4NF)
Transform fourth normal form to fifth normal form (5NF).

21.Most Relational Database Designers will accept that in order to deal with the complexities of "real life" data modeling, a design needs to reach

Third normal form**.**

Boyce-Codd normal form (BCNF), fourth normal form (4NF) and fifth normal form (5NF) are rarely used in practice.

22.An atomic field is

"Atomic" means "cannot be divided or split in smaller parts". Applied to databases fields(columns) this means that a column should not contain more than one value. It should not compose or combine values that have a meaning of their own.

23.A relation is in 1NF if it doesn't contain any ____________?

First normal form (1NF) is a property of a relation in a relational database. A relation is in first normal form if and only if the domain of**each attribute contains only atomic (indivisible) values**, and the value of each attribute contains only a single value from that domain.The first definition of the term, in a 1971 conference paper by Edgar Codd, defined a relation to be in first normal form when none of its domains have any sets as elements.

First normal form enforces these criteria:

Eliminate repeating groups in individual tables. Create a separate table for each set of related data. I**dentify each set of related data with a primary key.**

NOTE: Not NULL values in primary key fields is a more general requirement of relational databases; not necessarily of the first normal form.

24.Second Normal Form is designed to

Second normal form (2NF) is a normal form used in database normalization. 2NF was originally defined by E.F. Codd in 1971.

A relation that is in first normal form (1NF) must meet additional criteria if it is to qualify for second normal form. Specifically: a relation is in 2NF if it is in 1NF and no non-prime attribute is dependent on any proper subset of any candidate key of the relation. A non-prime attribute of a relation is an attribute that is not a part of any candidate key of the relation.

Put simply, a relation is in 2NF if it is in 1NF and every non-prime attribute of the relation is dependent on the whole of every candidate key.

25.What is a transitive dependency?

In Database Management System, a transitive dependency is a functional dependency which holds by virtue of transitivity. A transitive dependency can occur only in a relation that has three or more attributes. Let A, B, and C designate three distinct attributes (or distinct collections of attributes) in the relation. Suppose all three of the following conditions hold:

A → B It is not the case that B → A B → C Then the functional dependency A → C (which follows from 1 and 3 by the axiom of transitivity) is a transitive dependency.

In database normalization, one of the important features of third normal form is that it excludes certain types of transitive dependencies. E.F. Codd, the inventor of the relational model, introduced the concepts of transitive dependency and third normal form in 1971.

26.What does 3NF ensure?

Third normal form (3NF) is a normal form that is used in normalizing a database design to reduce the duplication of data and ensure referential integrity by ensuring that (1) the entity is in second normal form, and (2) all the attributes in a table are determined only by the candidate keys of that relation and not by any non-prime attributes. 3NF was designed to improve database processing while minimizing storage costs.

3NF was originally defined by E.F. Codd in 1971.Codd's definition states that a table is in 3NF if and only if both of the following conditions hold:

1.The relation R (table) is in second normal form (2NF)

2.Every non-prime attribute of R is non-transitively dependent on every key of R.

A non-prime attribute of R is an attribute that does not belong to any candidate key of R.[3] A transitive dependency is a functional dependency in which X → Z (X determines Z) indirectly, by virtue of X → Y and Y → Z (where it is not the case that Y → X).

An approximation of Codd's definition of 3NFwas given by Bill Kent:

> [Every] non-key [attribute] must provide a fact about the key, the whole key, and nothing but the key.

27. Consider the following table which of the following affirmations applies the best:

ID(Key Field) | FIRSTNAME | LASTNAME | BIRTHDATE | GENDER | CITYOFBIRTH | STATE

The table is in the first normal form, but more than that is in the second normal form. It is not in the third normal form because the STATE (a non key attribute) is depended on the CITYOFBIRTH (another non key attribute). It is not correct to say that the table is not normalized. It is to an extent.

28.What is the name of a field in a table that relates to the identifier in another table?

In the context of relational databases, a foreign key is a field (or collection of fields) in one table that uniquely identifies a row of another table or the same table.In simpler words, the foreign key is defined in a second table, but it refers to the primary key or a unique key in the first table. For example, a table called Employee has a primary key called employee_id. Another table called Employee Details has a foreign key which references employee_id in order to uniquely identify the relationship between both tables.

29.A foreign key must____

Since the purpose of the foreign key is to identify a particular row of referenced table, it is generally required that**the foreign key is equal to the candidate key in some row of the primary table, or else have no value (the NULL value**).**This rule is called a referential integrity constraint**between the two tables.Because violations of these constraints can be the source of many database problems, most database management systems provide mechanisms to ensure that every non-null foreign key corresponds to a row of the referenced table.

30.Is it possible for a table to have more than one foreign key?

Yes, a table can have many foreign keys and only one primary key.

31.What do the lines between tables (or entities) on a database diagram mean?

Relationships between tables and their cardinality:One-to-one, Many-to-one or One-to-many and Many-to-many.

32.The mapping of relationship depends on

The type of relationship.

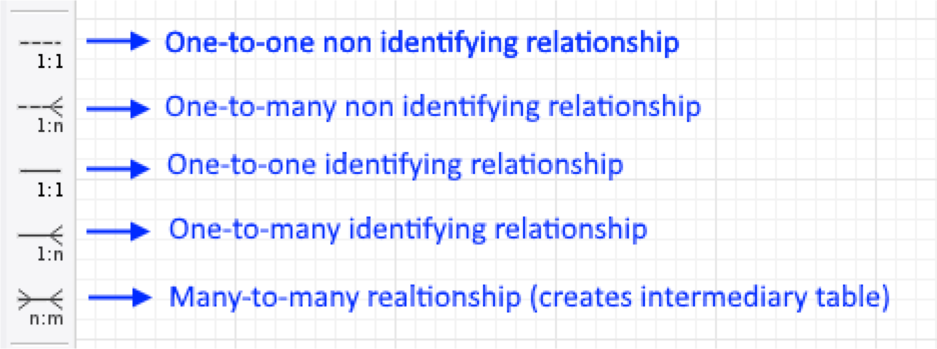

1:1
One-to-one non identifying relationship
1:n
One-to-many non identifying relationship
1:1
One-to-one identifying relationship
1:n
One-to-many identifying relationship
n:m
Many-to-many realtionship (creates intermediary table)

Fig. 1 - Relationship types and their graphical representation in MySQLWorkbench.

33.What are some valid types of relationships in a Relational Database?

There are several types of database relationships: One to One Relationships, One to Many and Many to One Relationships, Many to Many Relationships and Self Referencing Relationships.

34.A relationship that is rarely seen in real life is a

One-to-one relationship.

35.A one to many relationship (table A is in a one to many relationship with table B) is

In a relational database, a one-to-many relationship exists when one row in table A may be linked with many rows in table B, but one row in table B is linked to only one row in table A. It is important to note that a one-to-many relationship is not a property of the data, but rather of the relationship itself.For instance, think of A as mothers, and B as children. A mother can have several children, but a child can have only one biological mother.

The opposite of one-to-many is many-to-one.

36.Is a many-to-one relationship the same as a one-to-many?

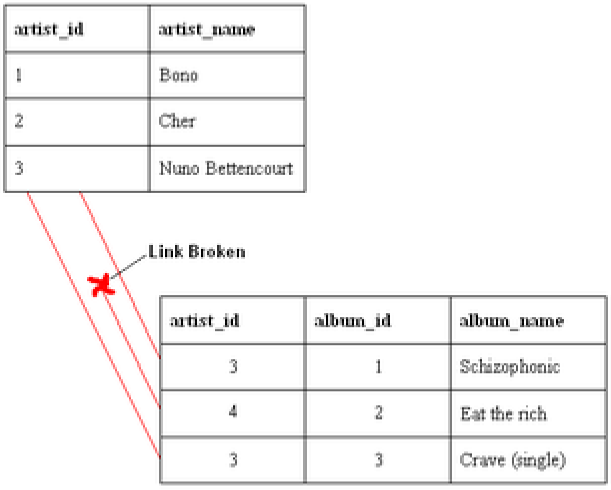
Yes. It depends on which side of the relationship the entity is present on.

37.Referential integrity refers to rules that a RDBMS will enforce. What are these rules?

Referential integrity is a property of data stating references within it are valid. In the context of relational databases, it requires every value of one attribute (column) of a relation (table) to exist as a value of another attribute (column) in a different (or the same) relation (table).

For referential integrity to hold in a relational database, any column in a base table that is declared a foreign key can contain either a null value, or only values from a parent table's primary key or a candidate key.In other words, when a foreign key value is used it must reference a valid, existing primary key in the parent table. For instance, deleting a record that contains a value referred to by a foreign key in another table would break referential integrity. Some relational database management systems (RDBMS) can enforce referential integrity, normally either by deleting the foreign key rows as well to maintain integrity, or by returning an error and not performing the delete. Which method is used may be determined by a referential integrity constraint defined in a data dictionary.

The adjective 'referential' describes the action that a foreign key performs, 'referring' to a link column in another table. In simple terms, 'referential integrity' is a guarantee that the target it 'refers' to will be found. A lack of referential integrity in a database can lead relational databases to return incomplete data, usually with no indication of an error.



artist_id	artist_name
1	Bono
2	Cher
3	Nuno Bettencourt

artist_id	album_id	album_name
3	1	Schizophonic
4	2	Eat the rich
3	3	Crave (single)

Fig. 2 -An example of a database that has not enforced referential integrity.

38.Which type of relationship uses a junction (intermediary) table?

In systems analysis, a many-to-many relationship is a type of cardinality that refers to the relationship between two entitiesA and B in which A may contain a parent instance for which there are many children in B and vice versa.

For example, think of A as Authors, and B as Books. An Author can write several Books, and a Book can be written by several Authors.

In a relational database management system, such relationships are usually implemented by means of an associative table (also known as cross-reference table), say, AB with two one-to-many relationships A -> AB and B -> AB. In this case the logical primary key for AB is formed from the two foreign keys (i.e. copies of the primary keys of A and B).

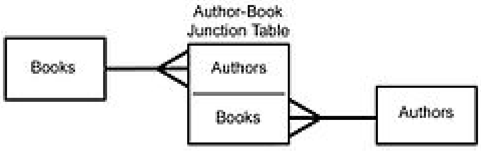


Fig. 3 - Many-to-many relationship using a junction table.

39.What is a many to many relationship

The same answer as for Question 34. Note:a many-to-many relationship cannot be represented directly in a relational database. You must use an intermediary (or junction, if you prefer this naming) table.