

DBMS - Interview Questions and Answers

Level 1

1. What is database?

A database is a logically coherent collection of data with some inherent meaning, representing some aspect of real world and which is designed, built and populated with data for a specific purpose.

2. What is DBMS?

It is a collection of programs that enables user to create and maintain a database. In other words it is general-purpose software that provides the users with the processes of defining, constructing and manipulating the database for various applications.

3. What is a Database system?

The database and DBMS software together is called as Database system.

4. What are the advantages of DBMS?

1. Redundancy is controlled.
2. Unauthorised access is restricted.
3. Providing multiple user interfaces.
4. Enforcing integrity constraints.
5. Providing backup and recovery.

5. What are the disadvantage in File Processing System?

1. Data redundancy and inconsistency.
2. Difficult in accessing data.
3. Data isolation.
4. Data integrity.
5. Concurrent access is not possible.
6. Security Problems.

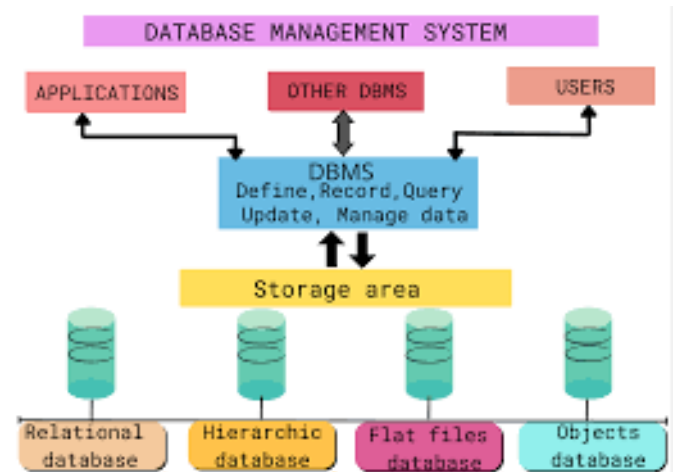
6. Describe the three levels of data abstraction?

The are three levels of abstraction:

1. **Physical level:** The lowest level of abstraction describes how data are stored.
2. **Logical level:** The next higher level of abstraction, describes what data are stored in database and what relationship among those data.
3. **View level:** The highest level of abstraction describes only part of entire database.

7. Define the "integrity rules"?

There are two Integrity rules.



1. **Entity Integrity:** States that "Primary key cannot have NULL value"
2. **Referential Integrity:** States that "Foreign Key can be either a NULL value or should be Primary Key value of other relation.

8. What is extension and intension?

1. **Extension:** It is the number of tuples present in a table at any instance. This is time dependent.
2. **Intension:** It is a constant value that gives the name, structure of table and the constraints laid on it.

9. What is System R? What are its two major subsystems?

System R was designed and developed over a period of 1974-79 at IBM San Jose Research Center. It is a prototype and its purpose was to demonstrate that it is possible to build a Relational System that can be used in a real life environment to solve real life problems, with performance at least comparable to that of existing system.

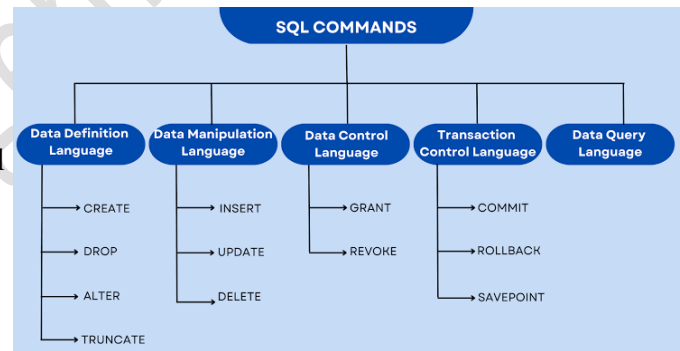
Its two subsystems are

1. Research Storage
2. System Relational Data System.

10. How is the data structure of System R different from the relational structure?

Unlike Relational systems in System R

1. Domains are not supported
2. Enforcement of candidate key uniqueness is optional
3. Enforcement of entity integrity is optional
4. Referential integrity is not enforced



11. What is Data Independence?

Data independence means that "the application is independent of the storage structure and access strategy of data". In other words, The ability to modify the schema definition in one level should not affect the schema definition in the next higher level.

Two types of Data Independence:

1. **Physical Data Independence:** Modification in physical level should not affect the logical level.
2. **Logical Data Independence:** Modification in logical level should affect the view level.

NOTE: Logical Data Independence is more difficult to achieve

12. What is a view? How it is related to data independence?

A view may be thought of as a virtual table, that is, a table that does not really exist in its own right but is instead derived from one or more underlying base table. In other words, there is no stored file that direct represents the view instead a definition of view is stored in data dictionary.

Growth and restructuring of base tables is not reflected in views. Thus the view can insulate users from the effects of restructuring and growth in the database. Hence accounts for logical data independence.

13. What is Data Model?

A collection of conceptual tools for describing data, data relationships data semantics and constraints.

14. What is E-R model?

This data model is based on real world that consists of basic objects called entities and of relationship among these objects. Entities are described in a database by a set of attributes.

15. What is Object Oriented model?

This model is based on collection of objects. An object contains values stored in instance variables with in the object. An object also contains bodies of code that operate on the object. These bodies of code are called methods. Objects that contain same types of values and the same methods are grouped together into classes.

16. What is an Entity?

It is a 'thing' in the real world with an independent existence.

17. What is an Entity type?

It is a collection (set) of entities that have same attributes.

18. What is an Entity set?

It is a collection of all entities of particular entity type in the database.

19. What is an Extension of entity type?

The collections of entities of a particular entity type are grouped together into an entity set.

20. What is Weak Entity set?

An entity set may not have sufficient attributes to form a primary key, and its primary key compromises of its partial key and primary key of its parent entity, then it is said to be Weak Entity set.

21. What is an attribute?

It is a particular property, which describes the entity.

22. What is a Relation Schema and a Relation?

A relation Schema denoted by $R(A_1, A_2, \dots, A_n)$ is made up of the relation name R and the list of attributes A_i that it contains. A relation is defined as a set of tuples. Let r be the relation which contains set tuples $(t_1, t_2, t_3, \dots, t_n)$. Each tuple is an ordered list of n -values $t=(v_1, v_2, \dots, v_n)$.

23. What is degree of a Relation?

It is the number of attribute of its relation schema.

24. What is Relationship?

It is an association among two or more entities.

25. What is Relationship set?

The collection (or set) of similar relationships.

26. What is Relationship type?

Relationship type defines a set of associations or a relationship set among a given set of entity types.

27. What is degree of Relationship type?

It is the number of entity type participating.

28. What is DDL (Data Definition Language)?

A data base schema is specifies by a set of definitions expressed by a special language called DDL.

29. What is VDL (View Definition Language)?

It specifies user views and their mappings to the conceptual schema.

30. What is SDL (Storage Definition Language)?

This language is to specify the internal schema. This language may specify the mapping between two schemas.

31. What is Data Storage - Definition Language?

The storage structures and access methods used by database system are specified by a set of definition in a special type of DDL called data storage-definition language.

32. What is DML (Data Manipulation Language)?

This language that enable user to access or manipulate data as organised by appropriate data model.

1. **Procedural DML or Low level:** DML requires a user to specify what data are needed and how to get those data.
2. **Non-Procedural DML or High level:** DML requires a user to specify what data are needed without specifying how to get those data.

33. What is DML Compiler?

It translates DML statements in a query language into low-level instruction that the query evaluation engine can understand.

34. What is Query evaluation engine?

It executes low-level instruction generated by compiler.

35. What is DDL Interpreter?

It interprets DDL statements and record them in tables containing metadata.

36. What is Record-at-a-time?

The Low level or Procedural DML can specify and retrieve each record from a set of records. This retrieve of a record is said to be Record-at-a-time.

37. What is Set-at-a-time or Set-oriented?

The High level or Non-procedural DML can specify and retrieve many records in a single DML statement. This retrieve of a record is said to be Set-at-a-time or Set-oriented.

38. What is Relational Algebra?

It is procedural query language. It consists of a set of operations that take one or two relations as input and produce a new relation.

39. What is Relational Calculus?

It is an applied predicate calculus specifically tailored for relational databases proposed by E.F. Codd. E.g. of languages based on it are DSL ALPHA, QUEL.

40. How does Tuple-oriented relational calculus differ from domain-oriented relational calculus?

1. The **tuple-oriented calculus** uses a tuple variables i.e., variable whose only permitted values are tuples of that relation. E.g. QUEL
2. The **domain-oriented calculus** has domain variables i.e., variables that range over the underlying domains instead of over relation. E.g. ILL, DEDUCE.

41. What is normalization?

It is a process of analysing the given relation schemas based on their Functional Dependencies (FDs) and primary key to achieve the properties

(1).Minimizing redundancy, (2). Minimizing insertion, deletion and update anomalies.

42. What is Functional Dependency?

A Functional dependency is denoted by $X \rightarrow Y$ between two sets of attributes X and Y that are subsets of R specifies a constraint on the possible tuple that can form a relation state r of R. The constraint is for any two tuples t1 and t2 in r if $t1[X] = t2[X]$ then they have $t1[Y] = t2[Y]$. This means the value of X component of a tuple uniquely determines the value of component Y.

43. What is Lossless join property?

It guarantees that the spurious tuple generation does not occur with respect to relation schemas after decomposition.

44. What is 1 NF (Normal Form)?

The domain of attribute must include only atomic (simple, indivisible) values.

45. What is Fully Functional dependency?

It is based on concept of full functional dependency. A functional dependency $X \twoheadrightarrow Y$ is full functional dependency if removal of any attribute A from X means that the dependency does not hold any more.

46. What is 2NF?

A relation schema R is in 2NF if it is in 1NF and every non-prime attribute A in R is fully functionally dependent on primary key.

47. What is 3NF?

A relation schema R is in 3NF if it is in 2NF and for every FD $X \twoheadrightarrow A$ either of the following is true

1. X is a Super-key of R .
2. A is a prime attribute of R .

In other words, if every non prime attribute is non-transitively dependent on primary key.

48. What is BCNF (Boyce-Codd Normal Form)?

A relation schema R is in BCNF if it is in 3NF and satisfies an additional constraint that for every FD $X \twoheadrightarrow A$, X must be a candidate key.

49. What is 4NF?

A relation schema R is said to be in 4NF if for every Multivalued dependency $X \twoheadrightarrow Y$ that holds over R , one of following is true.

- 1.) X is subset or equal to (or) $XY = R$.
- 2.) X is a super key.

50. What is 5NF?

A Relation schema R is said to be 5NF if for every join dependency $\{R_1, R_2, \dots, R_n\}$ that holds R , one the following is true 1.) $R_i = R$ for some i .

- 2.) The join dependency is implied by the set of FD, over R in which the left side is key of R .

51. What is Domain-Key Normal Form?

A relation is said to be in DKNF if all constraints and dependencies that should hold on the the constraint can be enforced by simply enforcing the domain constraint and key constraint on the relation.

52. What are partial, alternate,, artificial, compound and natural key?

1. **Partial Key:** It is a set of attributes that can uniquely identify weak entities and that are related to same owner entity. It is sometime called as Discriminator.
2. **Alternate Key:** All Candidate Keys excluding the Primary Key are known as Alternate Keys.

3. **Artificial Key:** If no obvious key, either stand alone or compound is available, then the last resort is to simply create a key, by assigning a unique number to each record or occurrence. Then this is known as developing an artificial key.
4. **Compound Key:** If no single data element uniquely identifies occurrences within a construct, then combining multiple elements to create a unique identifier for the construct is known as creating a compound key.
5. **Natural Key:** When one of the data elements stored within a construct is utilized as the primary key, then it is called the natural key.

53. What is indexing and what are the different kinds of indexing?

Indexing is a technique for determining how quickly specific data can be found.

Types:

1. Binary search style indexing
2. B-Tree indexing
3. Inverted list indexing
4. Memory resident table
5. Table indexing

54. What is system catalog or catalog relation? How is better known as?

A RDBMS maintains a description of all the data that it contains, information about every relation and index that it contains. This information is stored in a collection of relations maintained by the system called metadata. It is also called data dictionary.

55. What is meant by query optimization?

The phase that identifies an efficient execution plan for evaluating a query that has the least estimated cost is referred to as query optimization.

56. What is durability in DBMS?

Once the DBMS informs the user that a transaction has successfully completed, its effects should persist even if the system crashes before all its changes are reflected on disk. This property is called durability.

57. What do you mean by atomicity and aggregation?

1. **Atomicity:** Either all actions are carried out or none are. Users should not have to worry about the effect of incomplete transactions. DBMS ensures this by undoing the actions of incomplete transactions.
2. **Aggregation:** A concept which is used to model a relationship between a collection of entities and relationships. It is used when we need to express a relationship among relationships.

58. What is a Phantom Deadlock?

In distributed deadlock detection, the delay in propagating local information might cause the deadlock detection algorithms to identify deadlocks that do not really exist. Such situations are called phantom deadlocks and they lead to unnecessary aborts.

59. What is a checkpoint and When does it occur?

A Checkpoint is like a snapshot of the DBMS state. By taking checkpoints, the DBMS can reduce the amount of work to be done during restart in the event of subsequent crashes.

60. What are the different phases of transaction?

Different phases are

- 1.) Analysis phase,
- 2.) Redo Phase,
- 3.) Undo phase.

61. What do you mean by flat file database?

It is a database in which there are no programs or user access languages. It has no cross-file capabilities but is user-friendly and provides user-interface management.

62. What is "transparent DBMS"?

It is one, which keeps its Physical Structure hidden from user.

63. What is a query?

A query with respect to DBMS relates to user commands that are used to interact with a data base. The query language can be classified into data definition language and data manipulation language.

64. What do you mean by Correlated subquery?

Subqueries, or nested queries, are used to bring back a set of rows to be used by the parent query. Depending on how the subquery is written, it can be executed once for the parent query or it can be executed once for each row returned by the parent query. If the subquery is executed for each row of the parent, this is called a correlated subquery.

A correlated subquery can be easily identified if it contains any references to the parent subquery columns in its WHERE clause. Columns from the subquery cannot be referenced anywhere else in the parent query. The following example demonstrates a non-correlated subquery.

Example: `Select * From CUST Where '10/03/1990' IN (Select ODATE From ORDER Where CUST.CNUM = ORDER.CNUM)`

65. What are the primitive operations common to all record management systems?

Addition, deletion and modification.

66. Name the buffer in which all the commands that are typed in are stored?

'Edit' Buffer.

67. What are the unary operations in Relational Algebra?

PROJECTION and SELECTION.

68. Are the resulting relations of PRODUCT and JOIN operation the same?

No.

PRODUCT: Concatenation of every row in one relation with every row in another.

JOIN: Concatenation of rows from one relation and related rows from another.

69. What is RDBMS KERNEL?

Two important pieces of RDBMS architecture are the kernel, which is the software, and the data dictionary, which consists of the system-level data structures used by the kernel to manage the database. You might think of an RDBMS as an operating system (or set of subsystems), designed specifically for controlling data access; its primary functions are storing, retrieving, and securing data. An RDBMS maintains its own list of authorized users and their associated privileges; manages memory caches and paging; controls locking for concurrent resource usage; dispatches and schedules user requests; and manages space usage within its table-space structures.

70. Name the sub-systems of a RDBMS.

I/O, Security, Language Processing, Process Control, Storage Management, Logging and Recovery, Distribution Control, Transaction Control, Memory Management, Lock Management.

71. Which part of the RDBMS takes care of the data dictionary? How?

Data dictionary is a set of tables and database objects that is stored in a special area of the database and maintained exclusively by the kernel.

72. What is the job of the information stored in data-dictionary?

The information in the data dictionary validates the existence of the objects, provides access to them, and maps the actual physical storage location.

73. How do you communicate with an RDBMS?

You communicate with an RDBMS using Structured Query Language (SQL).

74. Define SQL and state the differences between SQL and other conventional programming Languages.

SQL is a nonprocedural language that is designed specifically for data access operations on normalized relational database structures. The primary difference between SQL and other conventional programming languages is that SQL statements specify what data operations should be performed rather than how to perform them.

75. Name the three major set of files on disk that compose a database in Oracle.

There are three major sets of files on disk that compose a database. All the files are binary. These are

- 1.) Database files
- 2.) Control files
- 3.) Redo logs

The most important of these are the database files where the actual data resides. The control files and the redo logs support the functioning of the architecture itself. All three sets of files must be present, open, and available to Oracle for any data on the database to be useable. Without these files, you cannot access the database, and the database administrator might have to recover some or all of the database using a backup, if there is one.

76. What is database Trigger?

A database trigger is a PL/SQL block that can be defined to automatically execute for insert, update, and delete statements against a table. The trigger can be defined to execute once for the entire statement or once for every row that is inserted, updated, or deleted. For any one table, there are twelve events for which you can define database triggers. A database trigger can call database procedures that are also written in PL/SQL.

77. What are stored-procedures? And what are the advantages of using them?

Stored procedures are database objects that perform a user defined operation. A stored procedure can have a set of compound SQL statements. A stored procedure executes the SQL commands and returns the result to the client. Stored procedures are used to reduce network traffic.

78. What is Storage Manager?

It is a program module that provides the interface between the low-level data stored in database, application programs and queries submitted to the system.

79. What is Buffer Manager?

It is a program module, which is responsible for fetching data from disk storage into main memory and deciding what data to be cache in memory.

80. What is Transaction Manager?

It is a program module, which ensures that database, remains in a consistent state despite system failures and concurrent transaction execution proceeds without conflicting.

81. What is File Manager?

It is a program module, which manages the allocation of space on disk storage and data structure used to represent information stored on a disk.

82. What is Authorization and Integrity manager?

It is the program module, which tests for the satisfaction of integrity constraint and checks the authority of user to access data.

83. What are stand-alone procedures?

Procedures that are not part of a package are known as stand-alone because they are independently defined. A good example of a stand-alone procedure is one written in a SQL*Forms application. These types of procedures are not available for reference from other Oracle tools. Another limitation of stand-alone procedures is that they are compiled at run time, which slows execution.

84. What are cursors give different types of cursors?

PL/SQL uses cursors for all database information accesses statements. The language supports the use two types of cursors

- 1.) Implicit
- 2.) Explicit

85. What is cold backup and hot backup (in case of Oracle)?

1. **Cold Backup:** It is copying the three sets of files (database files, redo logs, and control file) when the instance is shut down. This is a straight file copy, usually from the disk directly to tape. You must shut down the instance to guarantee a consistent copy. If a cold backup is performed, the only option available in the event of data file loss is restoring all the files from the latest backup. All work performed on the database since the last backup is lost.
2. **Hot Backup:** Some sites (such as worldwide airline reservations systems) cannot shut down the database while making a backup copy of the files. The cold backup is not an available option.

86. What is meant by Proactive, Retroactive and Simultaneous Update.

1. **Proactive Update:** The updates that are applied to database before it becomes effective in real world.
 2. **Retroactive Update:** The updates that are applied to database after it becomes effective in real world.
 3. **Simulatneous Update:** The updates that are applied to database at the same time when it becomes effective in real world.
-

Level 2

1. What are data and information, and how are they related in a database?

Data is recorded facts and figures, and information is knowledge derived from data. A database stores data in such a way that information can be created.

2. What is Enterprise Resource Planning (ERP), and what kind of a database is used in an ERP application?

Enterprise Resource Planning (ERP) is an information system used in manufacturing companies and includes sales, inventory, production planning, purchasing and other business functions. An ERP system typically uses a multiuser database.

3. What is a DBMS?

DBMS stands for Database Management System. A DBMS receives requests from applications and translates those requests into actions on a specific database. A DBMS processes SQL statements or uses other functionality to create, process and administer databases.

4. Why is a database considered to be "self-describing"?

In addition to the users' data, a database contains a description of its own structure. This descriptive data is called "metadata."

5. Who is E.F. Codd, and why is he significant in the development of modern database systems?

While working at IBM, E.F. Codd created the relational database model. A paper he published in 1970 presented his ideas to the world at large. His work is the foundation for most of the DBMSs currently in use, and thus forms the basis for database systems as we know and use them today.

6. What is SQL, and why is it important?

SQL stands for Structured Query Language, and is the most important data processing language in use today. It is not a complete programming language like Java or C#, but a data sublanguage used for creating and processing database data and metadata. All DBMS products today use SQL.

7. Write an SQL SELECT statement to display all the columns of the STUDENT table but only those rows where the Grade column is greater than or equal to 90.

```
SELECT * FROM STUDENT WHERE Grade >= 90;
```

8. Name and briefly describe the five SQL built-in functions.

COUNT: computes the number of rows in a table. SUM: totals numeric columns. AVG: computes the average value. MAX: obtains the maximum value of a column in a table. MIN: obtains the minimum value of a column in a table.

9. Write an SQL SELECT statement to count the number of rows in STUDENT table and display the result with the label NumStudents.

```
SELECT COUNT(*) AS NumStudents FROM STUDENT;
```

10. What is an SQL subquery?

An SQL subquery is a means of querying two or more tables at the same time. The subquery itself is an SQL SELECT statement contained within the WHERE clause of another SQL SELECT statement, and separated by being enclosed in parenthesis. Some subqueries have equivalent SQL join structures, but correlated subqueries cannot be duplicated by a join..

11. Discuss the alternative terminology that is used in the relational model.

Relations are also called tables, and sometimes by the older data processing term files. A row is known as a tuple in the relational model, but may also be referred to as a record. Finally, relational model attributes are known as table columns and sometimes as fields.

12. Why are functional dependencies not equations?

Equations deal with numerical relationships. A functional dependency deals with the existence of a determinant relationship between attributes, regardless of whether or not there is a numerical relationship between them.

Thus, if we know that there is no hot water every Wednesday, No-Hot-Water is functionally dependent on Wednesday. So, if we know it is Wednesday, then we know we will have No-Hot-Water. This is a functional dependency, but not an equation.

13. What is a foreign key, and what is it used for?

A foreign key is used to establish relationships among relations in the relational model. Technically, a foreign key is a column (or columns) appearing in one relation that is (are) the primary key of another table. Although there may be exceptions, the values in the foreign key columns usually must correspond to values existing in the set of primary key values. This correspondence requirement is created in a database using a referential integrity constraint on the foreign key.

14. What are insertion and deletion anomalies?

A deletion anomaly occurs when, by deleting the facts about one entity, we inadvertently delete facts about another entity; with one deletion, we lose facts about two entities. For example, if we delete the tuple for Student 001289 from a table, we may lose not only the fact that Student 001289 is in Pierce Hall, but also the fact that he has \$200 left in his security deposit. An insertion anomaly happens when we encounter the restriction that we cannot insert a fact about one entity until we have an additional fact about another entity. For example, we want to store the fact that the security deposit for Pierce Hall is \$300, but we cannot enter this data into the Student relation until a student registers for Pierce Hall.

15. What does it mean when we say that a relation is in Boyce-Codd Normal Form (BCNF)?

A relation is in BCNF when every determinant in the relation is a candidate key. This means that any possible primary key can determine all other attributes in the relation. Attributes may not be determined by non-candidate key attributes or part of a composite candidate key. Thus it is said "I swear to construct my tables so that all nonkey columns are dependent on the key, the whole key and nothing but the key, so help me Codd!"

16. You have been given a set of tables with data and asked to create a new database to store them. When you examine the data values in the tables, what are you looking for?

(1) Multivalued dependencies, (2) Functional dependencies, (3) Candidate keys, (4) Primary keys and (5) Foreign keys.

17. Why do normalized tables require more complex SQL when SQL statements are used in application programs?

Tables that are normalized contain data that has been distributed among the tables, but which may need to be recombined to answer queries from an application. To recombine the data, the programmer will have to use subqueries and/or joins. These SQL structures are more complex to write than a simple SELECT statement.

18. What is the multivalued, multicolumn problem? Include an example not used in the text.

The multivalued, multicolumn problem occurs when a table is designed to include multiple columns that hold variations of one type of attribute data. One example is where boat owners have the names of their boats stored as BOAT_01, BOAT_02 and BOAT_03.

19. Why is the multivalued, multicolumn problem another form of the multivalued dependency problem?

Both problems try to store multiple values on an attribute in a table. In the multivalued, multiplecolumn problem, the multiple values are stored in different columns. In the multivalued dependency problem the multiple values are stored in different rows. In both cases, the solution is the same: store the multiple values in a separate table.

20. What is the inconsistent values problem? Include an example not used in the text.

The inconsistent values problem occurs when different users or data sources use slightly different forms of the same data value. One example is where automobiles are specified as "Ford, 2-door, Red" in one cell and "Red Ford 2-door" in another.

21. Explain the relationship between entity, entity class, and entity instance.

An entity is something that can be identified in the users' work environment, something that the users want to track. Entities of a given type are grouped into entity classes. An entity instance is the representation of a particular entity.

22. Explain the difference between attributes and identifiers.

Entities have attributes. Attributes are properties that describe the entity's characteristics. Entity instances have identifiers. Identifiers are attributes that name, or identify, entity instances.

23. Name and describe three types of binary relationships.

1:1 - a single entity instance of one type is related to a single-entity instance of another type.

1:N - a single entity instance of one type is related to many-entity instances of another type.

M:N - many-entity instances of one type relate to many-entity instances of another type.

24. What is the archetype/instance pattern?

The archetype/instance pattern occurs when one entity tracks occurrences of another entity. A common example is the relationship between CLASS and SECTION, where the actual occurrence of a class happens when students register for a SECTION of that CLASS. The archetype/instance pattern is implemented as an ID-dependent relationship.

25. What is a recursive relationship? Give an example not used in the text.

A recursive relationship is a relationship between an entity and itself. For example, given the entity PERSON, a recursive relationship could be used to show a PERSON and his or her SIBLINGS (brothers and sisters).

26. What are the steps for transforming an entity into a table?

The steps are: (1) specify the primary key, (2) specify candidate keys, (3) specify column properties including null status, data type, default value (if any), and data constraints (if any), and (4) verifying normalization.

27. Define a surrogate key, describe the ideal primary key and explain how surrogate keys meet this ideal.

The ideal primary key is short, numeric and fixed. A surrogate key is a unique, DBMS-supplied identifier intended to be used as the primary key of a table. Further, the DBMS will not allow the value of a surrogate key

to be changed. The values of a surrogate key have no meaning to the users and are usually hidden on forms and reports. By design, they are short, numeric and fixed and thus meet the definition of the ideal primary key

28. Define and discuss data constraints.

Data constraints on a column are the limits put on the values the data can have. There are four types of data constraints: (1) domain constraints, which define a limited set of values for the column, (2) range constraints, which specify that the values must fall within a certain range, (3) intrarelation constraints, which define what values the column can have based on values of other columns in the same table, and (4) interrelation constraints, which define values the column can have based on values of columns in other tables.

29. In general, how are recursive relationships handled in a database design?

A recursive relationship is a relationship among entities of the same class, and is represented in the same way as other relationships are. The rows of the tables can take two different roles, however. Some are parent rows, and others are child rows. Further, the table will contain both its own primary key and the foreign key that links back to the table itself. If a row has no parent, then the value of the foreign key column in that row will be null. If the row has a parent, then there must be a foreign key value in that row that corresponds to the primary key value of another row in the table.

30. What is a cascading update?

Referential integrity constraints require that foreign key values in one table correspond to primary key values in another. If the value of the primary key is changed -- that is, updated -- the value of the foreign key must immediately be changed to match it. Cascading updates will set this change to be done automatically by the DBMS whenever necessary.

31. What is a SQL view? Briefly explain the use of views.

A SQL view is a virtual table built from other tables or views. Views are used to (1) hide columns or rows, (2) show the results of computed columns, (3) hide complicated SQL syntax, (4) layer built-in functions, (5) provide a level of indirection between application programs and tables, (6) assign different sets of processing permissions to tables, and (7) to assign different sets of triggers to the same table.

32. Explain the "paradigm mismatch" between SQL and application programming languages.

SQL statements return a set of rows, while an application program works on one row at a time. To resolve this mismatch the results of SQL statements are processed as pseudofiles, using a cursor or pointer to specify which row is being processed.

33. Name four applications for triggers.

(1) providing default values, (2) enforcing data constraints, (3) updating views and (4) enforcing referential integrity

34. What are stored procedures, and how do they differ from triggers?

A stored procedure is a program that is stored within the database and is compiled when used. They can receive input parameters and they can return results. Unlike triggers, their scope is database-wide; they can be used by any process that has permission to use the database stored procedure.

35. What are the advantages of using stored procedures?

The advantages of stored procedures are (1) greater security, (2) decreased network traffic, (3) the fact that SQL can be optimized and (4) code sharing which leads to less work, standardized processing, and specialization among developers.

36. Why is database redesign necessary?

Database redesign is necessary for two reasons. First, redesign is necessary both to fix mistakes made during the initial database design. Second, redesign is necessary to adapt the database to changes in system requirements. Such changes are common because information systems and organizations do not just influence each other they create each other. Thus, new information systems cause changes in systems requirements.

37. What is the difference between a correlated subquery and a regular subquery?

A correlated subquery appears deceptively similar to a regular subquery. The difference is that a regular subquery can be processed from the bottom up. In a regular subquery, results from the lowest query can be determined and used to evaluate the upper-level query. In contrast, in a correlated subquery, the processing is nested; that is, a row from an upper query statement is used in comparison with rows in a lower level query. The key distinction of a correlated subquery is that the lower-level select statements use columns from upper-level statements.

38. What is a dependency graph?

A dependency graph is a diagram that is used to portray the connections between database elements. For example, a change in a table can potentially impact relationships, views, indexes, triggers, stored procedures, and application programs. These impacts need to be known and accounted for before making database changes.

39. Explain how to add a NOT NULL column to a table.

First, add the column as NULL. Then use UPDATE to add data to every row. Finally use an ALTER TABLE . . . ALTER COLUMN statement to change the column constraint to NOT NULL.

40. You have two tables, EMPLOYEE and COMPUTER that are in a one-to-one relationship. The foreign key is EmpNumber in COMPUTER which references EmpNumber as the primary key of EMPLOYEE. Explain what must be done to convert the one-to-one EMPLOYEE-COMPUTER relationship to a one-to-many relationship where one employee can have more than one computer.

In the one-to-one relationship, there will be a constraint on EmpNumber as a foreign key in COMPUTER stating that EmpNumber must be unique. To convert the relationship to a one-to-many relationship, just drop this constraint.

41. Explain the difference between an exclusive lock and a shared lock.

An exclusive lock prohibits other users from reading the locked resource; a shared lock allows other users to read the locked resource, but they cannot update it.

42. Explain the difference between optimistic locking and pessimistic locking.

Optimistic locking assumes no transaction conflict will occur and deals with the consequences if it does. Pessimistic locking assumes that conflict will occur and so prevents it ahead of time with locks. In general, optimistic locking is preferred for the Internet and for many intranet applications.

43. What is deadlock? How can it be avoided? How can it be resolved once it occurs?

Deadlock occurs when two transactions are each waiting on a resource that the other transaction holds. Deadlock can be prevented by requiring transactions to acquire all locks at the same time; once it occurs, the only way to cure it is to abort one of the transactions and back out of partially completed work.

44. What are the major functions of the database administrator?

Managing database structure, controlling concurrent processing, managing processing rights and responsibilities, developing database security, providing for database recovery, managing the DBMS and maintaining the data repository.

45. Explain what we mean by an ACID transaction.

An ACID transaction is one that is atomic, consistent, isolated, and durable. Durable means that database changes are permanent. Consistency can mean either statement level or transaction level consistency. With transaction level consistency, a transaction may not see its own changes. There are four transaction isolation levels: read committed, read uncommitted, repeatable read and serialization. Atomic means it is performed as a unit.

46. What are the ways in which an Oracle database can be created?

There are three (3) ways to create an Oracle database. You can create a database using the Database Configuration Assistant, using the Oracle-supplied database creation procedures or using the SQL CREATE DATABASE command.

47. What are sequences, and what are the possible problems when using them to create surrogate keys?

A sequence is an object that generates a sequential series of unique numbers. Sequences are most often used to provide values for surrogate keys. However, there are three problems with using sequences. First, a developer can use a defined sequence for any purpose; and if a sequence is used for purposes other than the surrogate key, some values will be missing. A second problem is that there is nothing in the schema that prevents someone from issuing an INSERT statement that adds data to the table but that does not use the sequence. Finally, it is possible to use the wrong sequence when putting data into a table.

48. Under what conditions should indexes be used?

Indexes can be created to enforce uniqueness, to facilitate sorting, and to enable fast retrieval by column values. A good candidate for an index is a column that is frequently used with equal conditions in WHERE clauses.

49. Explain the three levels of transaction isolation supported by Oracle.

Oracle supports read committed, serializable, and read-only transaction isolation levels. Because of the way Oracle System Change Number (SCN) values are processed, Oracle never reads dirty data. Serializable isolation is possible, but the application program must be written to process the "Cannot serialize" exception. Applications can place locks explicitly using SELECT FOR UPDATE commands but this is not recommended.

50. What are the types of files used in Oracle recovery?

Datafiles, control files and two types of ReDo log files: OnLine Redo and Offline ReDo (which is also known as Archive ReDo).

51. What is the difference between SQL Server 2000 complete and differential backups?

A complete backup makes a copy of the entire database. A differential backup makes a copy of the changes that have been made to the database since the last complete backup. A complete backup must be made before the first differential backup. Because differential backups are faster, they can be taken more frequently and the chance of data loss is reduced. Complete backups take longer but are slightly simpler to use for recovery.

52. Explain the meaning of each of the transaction levels supported by SQL Server.

The strictest isolation level is **SERIALIZABLE**. With it, SQL Server places a range lock on the rows that have been read. This level is the most expensive to use and should only be used when absolutely required. The next most restrictive level is **REPEATABLE READ**, which means SQL Server places and holds locks on all rows that are read. It is possible to make dirty reads by setting the isolation level to **READ UNCOMMITTED**, which is the least restrictive level. **READ COMMITTED** is the default isolation level.

53. Explain the difference between the SQL Server 2000 simple, full, and bulk-logged recovery models.

With the simple recovery model, no logging is done. The only way to recover a database is to restore the database to the last backup. With full recovery, all database changes are logged. With bulk-logged database recovery, all changes are logged except those that cause large log entries.

54. What is the difference between SQL Server 2000 clustered and nonclustered indexes?

With a clustered index, the data are stored in the bottom level of the index and in the same order as that index. With a nonclustered index, the bottom level of an index does not contain data; it contains pointers to the data. For data retrieval, clustered indexes are faster than nonclustered indexes.

55. What triggers does SQL Server 2000 support?

SQL Server 2000 supports **INSTEAD OF** and **AFTER** triggers only. There is no SQL Server support for **BEFORE** triggers. A table may have one or more **AFTER** triggers for insert, update and delete actions; **AFTER** triggers may not be assigned to views. A view or table may have at most one **INSTEAD OF** trigger for each triggering action of insert, update or delete.

56. What is the relationship of ODBC, OLE DB, and ADO?

Developed first, the ODBC standard is for relational databases; while the OLE DB standard provides functionality for both relational and other databases. Finally, ADO was developed to provide easier access to OLE DB data for the non-object-oriented programmer.

57. What are the three types of data sources used with ODBC?

An ODBC file data source is a file that can be shared among database users. A ODBC system data source is one that is local to a single computer. A ODBC user data source is only available to the user who created it.

58. What disadvantage of ODBC does OLE DB overcome?

By breaking the features and the functions of a DBMS into COM objects, OLE DB characteristic overcomes a major disadvantage of ODBC. With ODBC, a vendor must create an ODBC driver for almost all DBMS features and functions in order to participate in ODBC at all. This is a large task that requires a substantial initial investment. With OLE DB, however, a DBMS vendor can implement portions of their product.

59. What are to goals of OLE DB?

The major goals of OLE DB are to: (1) Create object interfaces for DBMS functionality pieces; (2) Increase flexibility for developers and users; (3) provide an object interface over any type of data; and (4) do not force data to be converted or moved from where it is.

60. In OLE DB, what is the difference between an interface and an implementation?

An OLE DB interface is specified by a set of objects, and the properties and methods that they expose, and OLE DB defines standardized interfaces. An object need not expose all of its properties and methods in a given interface. An OLE DB implementation defines how the object supports the interface. The implementation is completely hidden from the user. Thus developers of an object are free to change the implementation whenever they want, but they should not change the interface without consulting their users.

61. Why is XML a better markup language than HTML?

XML is a better markup language than HTML, primarily because XML provides a clear separation between document structure, content, and materialization. Symbols cannot be used ambiguously with XML.

62. What are the two means to describe the content of XML documents?

DTD (Document Type Declarations) and XML Schemas. An XML document that conforms to its DTD is called type-valid. A document can be well-formed and not be type-valid, either because it violates the structure of its DTD or because it has no DTD. However, DTDs have limitations and to overcome these limits XML Schemas were created. XML Schemas are XML documents that are the preferred method for defining document structure.

63. What is the difference between simple elements and complexType elements?

Simple elements have only one data value. ComplexType elements can have multiple elements nested within them. ComplexTypes may also have attributes. The elements contained in a complexType may be simple or other complexTypes. ComplexTypes may also define element sequences.

64. What is ADO.NET?

ADO.NET is a new, improved, and greatly expanded version of ADO that was developed for the Microsoft .NET initiative. ADO.NET incorporates all of the functionality of ADO, but adds much more. In particular, ADO.NET facilitates the transformation of XML documents to and from database data. Most importantly, ADO.NET introduces the concept of datasets, which are in-memory, full-function, independent databases.

65. What is a dataset?

A dataset is an in-memory database that is disconnected from any regular database, but has all the important characteristics of a regular database. Datasets can have multiple tables, relationships, referential integrity rules, referential integrity actions, views and the equivalent of triggers. Datasets are needed to provide a standardized, non-proprietary means to process database views. They are especially important for the processing of views with multiple multivalued paths.

66. Briefly describe the four JDBC driver types that Sun defines.

Type 1 drivers provide a bridge between Java and ODBC. Types 2-4 drivers are written entirely in Java, but differ as to how they connect to the DBMS. Type 2 drivers rely on the DBMS product for intermachine communication, if any. Type 3 drivers translate JDBC calls into a DBMS-independent network protocol. Type 4 drivers translate JDBC calls into a DBMS-dependent network protocol.

67. What is the difference between a Java servlet and a Java applet?

An applet is a compiled Java bytecode program that is transmitted to a browser via HTTP and is invoked using the HTTP protocol on the user's workstation. A servlet is a Java program that is invoked on the server to respond to HTTP requests.

68. What is the coding pattern for using a JDBC driver?

1. Load the driver.
2. Establish a connection to the database.
3. Create a statement.
4. Do something with the statement.

69. What is a Java bean?

A Java bean is a Java class that has the following three properties. (1) There are no public instance variables. (2) All persistent values are accessed using methods named getxxx and setxxx. (3) The bean class must either have no constructors or it must have one explicitly defined zero-argument constructor.

70. How are surrogate keys and metadata handled in MySQL?

MySQL uses integer data types combined with the property AUTO_INCREMENT to create surrogate keys. This creates a sequence that starts at one (1) and increases by one (1) for each new record. MySQL maintain its metadata in a database named mysql. For example, this database maintains two tables named user and db.

71. What is a data mart?

A data mart is a collection of data smaller in scope and size than a data warehouse. It is dedicated to data from a particular business component of business functional area. A data mart may function as a subset of a larger data warehouse. Users of a data mart are usually knowledgeable analysts in the business area using the data mart.

72. What is RFM analysis?

RFM analysis is a Business Intelligence (BI) reporting system that analyzes and ranks customers based on their purchasing patterns. R refers to "how recently" a customer placed an order, F refers to "how frequently" the customer orders, and M refers to "how much money" the customer spends. Typically, the customers are ranked into "20%" groups and assigned a number to represent their ranking. Thus 1 means top 20%, 2 the next 20% and so on. In this system a score of 1 is best and a score of 5 is worst. Thus a customer with an RFM score = 1 5 1 would be one who has ordered recently, does not order frequently, and who makes large purchases.

73. What are the functions of a reporting system?

A reporting system has three functions: 1. Report authoring -- connecting to data sources, creating the report structure and formatting the report. 2. Report management -- defining who receives which reports, when they receive them and how the reports are delivered. 3. Report delivery -- based on report management metadata, either pushing the reports to the recipients or allowing them to be pulled by the recipients.

74. What is OLAP?

OnLine Analytical Processing (OLAP) is a Business Intelligence (BI) reporting system. OLAP provides the user with the capability to sum, count, average and do other simple arithmetic operations on groups of data. An OLAP report has measures and dimensions. Measures are the data values to be displayed. Dimensions are characteristics of the measures. OLAP reports are called OLAP cubes, although such reports are not limited to three dimensions.

75. What is market basket analysis?

Market basket analysis is a data mining technique that determines which sets of products tend to be purchased together. A common technique uses conditional probabilities. In addition to the basic probability that an item will be purchased, three results are of particular interest:

Support -- the probability of two items being purchased together.

Confidence -- the probability of a second item being purchased GIVEN that another item has been purchased.

Lift -- calculated as confidence divided by a basic probability, this shows the likelihood of a second item being purchased IF an item is purchased.

76. Explain the differences between structured data and unstructured data.

Structured data are facts concerning objects and events. The most important structured data are numeric, character, and dates. Structured data are stored in tabular form. Unstructured data are multimedia data such as documents, photographs, maps, images, sound, and video clips. Unstructured data are most commonly found on Web servers and Web-enabled databases.

77. Explain why it is still necessary to have at least some familiarity with file processing systems even though it has become evident that traditional file processing systems have a number of shortcomings and limitations.

Many businesses still use file processing systems today. This is especially true in the creation of backups for a database system. In addition, if you understand some of the limitations of a file processing system such as program-data dependence, duplication of data, limited data sharing, lengthy development times, and excessive

program maintenance, you can try and avoid them as you design and develop a databases.

78. What are some of the disadvantages associated with conventional file processing systems?

There are five disadvantages. Program-data dependence occurs when file descriptions need to be changed in all programs whenever a file description changes. Duplication of data is storing the data more than one time. Limited data sharing occurs when the files are private so no one outside of one application can access the data. Lengthy development times exist because file processing systems takes longer to develop. Lastly, excessive program maintenance exists since the effort to maintain a program is larger in this environment.

79. The range of database applications can be divided into five categories. Explain the five different categories.

Databases can support from a single user (personal database) up to supporting the requests of the world (internet database). In between, a database can support a workgroup (a relatively small group of people), department database (a functional unit in an organization such as marketing), or an enterprise database (entire organization).

80. Explain the differences between an intranet and an extranet.

An Internet database is accessible by everyone who has access to a Web site. An intranet database limits access to only people within a given organization. An extranet database limits access to only people within a company and a company's customers and suppliers.

81. Briefly explain the five components of an Information Systems Architecture and their relationship to data.

Processes that manipulate the data. Networks that transport the data. People who process and use the data. Events and point in time when processes need to be performed with the data. Reasons for events and rules to govern the processing of the data.

82. Explain the systems development life cycle.

This is the traditional method to develop, maintain, and replace information systems that includes the project identification and selection, project initiation and planning, analysis, logical design, physical design, implementation, and maintenance steps. The process is a series of steps that are ideally completed in a linear fashion. In reality, the process is not linear and the process requires steps to be revisited and an overlap of steps.

83. Explain the differences of the two principal types of packaged data models.

Universal data models are common to many organizations. These models may be useful for similar functions that are used across companies or organizations such as purchasing and accounting. Industry-specific data models are used by specific industries.

84. Who can make up a systems or database team? Provide a brief explanation of each.

The team includes a system analyst who identifies the need for information services to meet opportunities of the business, database analysts who design the database, users who monitor that the system will meet their needs, programmers who write computer programs, database and data administrators who have responsibility

for current and future databases and other technical experts.

85. Briefly describe the six database activities that occur during the systems development life cycle.

The enterprise modeling that analyzes the current data processing. Conceptual data modeling that identifies entities, relationships, and attributes. The logical database design that identifies data integrity and security requirements. The physical database design and definition that defines the database to a DBMS. The database implementation that installs and converts data from prior systems. Database maintenance that fixes errors in the database and database applications.

86. Briefly explain an ERD.

An ERD is a detailed logical representation of the data for an organization. The ERD includes entities, attributes, relationships, and cardinalities. An ERD is the mechanism where an entity-relationship model is displayed.

87. List some of the characteristics of good data definitions.

Definitions are gathered from the same sources and should be accompanied diagrams. A definition will include special conditions, examples, how the data is created, whether the data can change, who owns the data, whether the data is optional, and whether the data can be broken into something more atomic.

88. Explain minimum and maximum cardinality.

Minimum cardinality is the minimum number of instances of an entity that can be associated with each instance of another entity. Maximum cardinality is the maximum number of instances of an entity that can be associated with each instance of another entity.

89. Describe the naming of Relationships?

Relationships are verb phrases and represent actions usually in the present tense. The name should exclude vague names such as "has". The name should explain what action is being taken and possibly why it is important.

90. Why is modeling time-dependent data with a time stamp important?

The values of data may change. A time stamp helps to ensure that the previous value of the data stays in the database after it has changed so that you can see the before and after values through time. Without a time stamp, you will most likely lose some of the history.

91. Explain the difference between total specialization and partial specialization.

Total specialization exists when every instance of a supertype must also be an instance of a subtype. Partial specialization exists when every instance of a supertype does not have to be an instance of a subtype.

92. Explain the difference between an ERD and EER.

An EER includes everything in an ERD and an EER allows for more complex relationships than an ERD. An EER allows for object-oriented data modeling and include supertypes and subtypes entities and inheritance.

93. Explain the difference between the disjoint and overlap rule.

The disjoint rule states an entity instance of a supertype can only be a member of one subtype. The overlap rule states an entity instance of a supertype can be a member of multiple subtypes.

94. List the three types of business rules and define each of them.

A derivation is a statement that is derived from other knowledge. A structured assertion is a statement that expresses some aspect of the static structure of an organization. An action assertion is a statement of a constraint on the actions of an organization.

95. Explain how a scenario is used for business rules.

A scenario is used to test business rules. It is a short script that describes how a business reacts to certain situations.

96. Explain some of the main goals of normalization.

Normalization should minimize data redundancy. It should also simplify referential integrity constraints. Normalization will also make it easier to insert, update, and delete data. And finally, it provides better design.

97. List some of the properties of a relation.

Relations in a database have a unique name and no multivalued attributes exist. Each row is unique and each attribute within a relation has a unique name. The sequence of both columns and rows is irrelevant.

98. Explain what needs to happen to convert a relation to third normal form

First you must verify that a relation is in both first normal form and second normal form. If the relation is not, you must convert into second normal form. After a relation is in second normal form, you must remove all transitive dependencies.

99. Describe how a supertype/subtype relationship is mapped into a relation.

A separate relation is created for each supertype and subtype. The attributes common for all of the subtypes are assigned to the supertype. Each subtype has the primary key from the supertype assigned to it. A subtype discriminator is added to the supertype.

100. Describe domain constraints.

Domain constraints include entity integrity and referential integrity. The domain is a set of values that may be assigned to an attribute. The entity integrity rule states that no part of a primary key cannot be null. Referential integrity states that each foreign key value must match a primary key value or be null.

101. What are the four objectives of the selection of a data type?

A data type should be selected so that all possible values are represented using minimal storage space. The data type should help to ensure data integrity and support all possible data manipulations (i.e., cannot place a letter in a field such as salary where a number is required).

102. Describe the four types of indexes.

A unique primary index is unique and is used to find and store a row. A nonunique primary index is not unique and is used to find a row but also where to store a row (based on its unique primary index). A unique secondary index is unique for each row and used to find table rows. A nonunique secondary index is not unique and used to find table rows.

103. What is denormalization and why would someone consider doing so?

Denormalization is the process of taking normalized relations and changing them so that they are not longer normalized. This process may lead to anomalies and create data redundancy as negative consequences. However, the revised relations should improve database performance.

104. Compare a hierarchical and network database model?

The hierarchical model is a top-down structure where each parent may have many children but each child can have only one parent. This model supports one-to-one and one-to-many relationships. The network model can be much more flexible than the hierarchical model since each parent can have multiple children but each child can also have multiple parents. This model supports one-to-one, one-to-many, and many-to-many relationships.

105. Describe the differences between vertical and horizontal partitioning.

Horizontal partitioning is where the rows in a relation are separated by some criteria and placed into a new relation or file with the same layout as the original relation (in this case only the records in each file differ). Vertical partitioning is where the columns in a relation are separated by some criteria and placed into a new relation or file with a different layout as the original relation.

106. Explain the difference between a dynamic and materialized view.

A dynamic view may be created every time that a specific view is requested by a user. A materialized view is created and or updated infrequently and it must be synchronized with its associated base table(s).

107. Discuss some of the techniques that can be used to tune operational performance.

Choosing primary and secondary keys can increase the speed of row selection, joining, and row ordering. Selecting the appropriate file organization for base tables and indexes can also improve performance. Clustering related rows together and maintaining statistics about tables and indexes can lead to increased efficiency.

108. Briefly describe the three types of SQL commands.

Data definition language commands are used to create, alter, and drop tables. Data manipulation commands are used to insert, modify, update, and query data in the database. Data control language commands help the DBA to control the database.

109. What are the steps to follow when preparing to create a table?

1. Identify the data type, length, and precision for each attribute. 2. Identify the columns that can accept a null value. 3. Identify the columns that need to be unique. 4. Identify primary and related foreign keys with the

parent table being created before the child. 5. Determine default values. 6. Determine where the domain values are that need to be constrained. 7. Create the indexes.

110. What are some disadvantages of a standard language such as SQL?

A standard language can hinder the effort to create a new language. One standard is never enough to meet all of the business needs. A standard can be a compromise among interested parties which can cause the standard to not be ideal. If a standard is altered by some, then portability between platforms could be hurt.

111. Explain a join between tables

A join allows tables to be linked to other tables when a relationship between the tables exists. The relationships are established by using a common column in the tables and often uses the primary/foreign key relationship.

112. Describe and contrast a trigger and a procedure.

Triggers are stored and controlled in the DBMS. A trigger is executed automatically when a condition is met (INSERT, UPDATE, or DELETE). A procedure is also stored in a database. A procedure is not executed automatically.

113. Briefly describe an outer join.

An outer join includes the records that match and those that do not have a matching value in another table. Outer joins can be a LEFT outer join (includes all records from the first table listed) or a RIGHT outer join (includes all records from the second table listed). Outer joins are not easily used with more than two tables.

114. Describe a subquery.

A subquery is a query that is composed of two queries. The first query (inner query) is within the WHERE clause of the other query (outer query). In some cases the inner query provides results for the outer query to process. In other cases, the outer query results provide results for the inner query (correlated subquery).

115. Describe the difference between embedded and dynamic SQL.

Embedded SQL is the process of including hard coded SQL statements. These statements do not change unless the source code is modified. Dynamic SQL is the process of generating SQL on the fly. The statements generated do not have to be the same each time.

116. Explain the difference between two and three-tier architectures.

A three-tier architecture includes a client and two server layers. The application code is stored on the application server and the database is stored on the database server. A two-tier architecture includes a client and one server layer. The database is stored on the database server.

117. Describe and contrast SQL and QBE.

QBE is a direct-manipulation database language that uses a graphical approach to query construction. Some database systems translate QBE queries into SQL. QBE does not adhere to a standard but SQL does. Both

SQL and QBE are relational database languages.

118. Describe ODBC

ODBC is a standard that contains an interface that provides a common language for application programs to access and process SQL databases. In order to use ODBC, a driver, server name, database name, user id, and password are required. ODBC is important for Internet applications and has gained wide acceptance.

119. Compare a thin and a fat client.

A fat client is a PC that is responsible where most processing occurs on the client rather than the server. Some of the fat client activities include processing, presentation logic, and business rules logic. A thin client is a PC where only limited processing occurs. In these cases, more processing should occur on the application server.

120. Describe some reasons that someone familiar with Microsoft Access will want to learn VBA.

You can perform more complex functions and error handling can be accommodated by VBA. VBA code will execute faster since code runs faster than macros and maintenance is easier because VBA modules are stored with the forms and reports. Reading VBA is easier because you can see the entire code. Finally, you can pass parameters and use OLE automation better.

121. Describe middleware applications that ease the connection of databases to Web applications.

Both ASP and ColdFusion are middleware that ease the connection of databases to Web applications. Both require the use of several files. Both use tags and are executed on the server. Both Internet Explorer and Netscape can access these files. The programmer does not need to be concerned with the client when they work with this middleware.

122. Describe JavaScript and VBScript.

JavaScript shares many features with Java. Developers use it to achieve interactivity. JavaScript is an open language and does not require a license. It is supported by both Internet Explorer and Netscape. VBScript is similar to JavaScript. It is based on Visual Basic but is simpler. Microsoft developed this language.

123. Describe Web Services.

Web Services are improving the ability of computers to communicate over the Internet. These services use XML programs and usually run in the background. Easier integration of applications may be possible because developers do not need to be familiar with the technical details with applications that are being integrated. UDDI is a technical specification for creating a distributed registry of Web services and businesses that are open to communicating through Web services.

124. Provide an overview of XML.

XML is used to structure and manipulate data involved with a browser and is becoming the standard for e-commerce. XML uses tags that are similar to HTML in that they use the angle brackets, but XML describes the content whereas HTML describes the appearance. The XML schema standard was published in May 2001 by W3C.

125. Describe Website security issues.

Website security issues include unauthorized access to the several aspects of one's Website. Security measures should include all aspects of the system such as the network, operating level, database, and Web server. Regular monitoring and security testing by a company should help to avoid intrusion into one's system.

126. Explain the role of metadata for the three-layer architecture.

Each of the three layers has a metadata layer linked with it. The metadata layer describes the properties or characteristics of the data. The operational metadata describe the data used in the various operational and external systems. The enterprise data warehouse metadata describe the reconciled data layer. The data mart metadata describes the derived data layer.

127. Describe why operational and informational systems are separate.

Operational systems are used to run the business in real time on a daily basis and contain current data. Non-managers and non-analysts work in this type of system. Operational systems must process large volumes of data. Informational systems are used to support decisions and contain historical data. Managers and analysts work in this type of system. Informational systems have periodic batch updates.

128. Describe a data warehouse.

A data warehouse is organized around specific subjects. The data is defined using consistent naming, format, and encoding structure standards. The data contains a time element, so that the data can be studied for trends. No data in a data warehouse can be updated by end users.

129. Explain why an information gap exists in most organizations.

One reason that an information gap exists is the fact that systems have been developed in separate, segmented efforts. This has helped the data from being stored in an integrated database and thus the data is in an inconsistent structure. The other reason for the gap is that most systems are created to support the operational aspect of an organization. The systems were not developed for decision making.

130. Describe the differences between a data warehouse and data mart.

A data warehouse is for the enterprise and contains multiple subjects. A data mart is for a specific functional area and focuses on one subject. A data warehouse is flexible and data-oriented and contains detailed historical data. A data mart is restrictive and project-oriented and contains some historical data.

131. Describe the difference between data and database administration.

Data administration is responsible for the overall management of data resources. Some of the core roles include the creation of data policies, procedures and standards, resolve data ownership issues, and manage the information repository. Database administration is physical database oversight. Some of the core duties include the selection of the DBMS and software tools, the installation and upgrade of the DBMS, and database performance tuning.

132. What are some of the important security features of a DBMS?

One of the features includes the use of views which allows the presentation of only data needed by someone

and limits the capability of database updates. The use of integrity controls includes such things as domains, assertions, and checks. Also authorization rules, user-defined procedures, encryption, authentication schemes, and backups are important.

133. Describe concurrency control.

Concurrency control is the process managing simultaneous operations against a database so that database integrity is not compromised. There are two approaches to concurrency control. The pessimistic approach involves locking and the optimistic approach involves versioning.

134. Explain locking.

Locking is denying others the ability to update a record until someone completes the update or releases the record. Locking can occur on many different levels in a database. It can occur at the database, table, record, or field level. A lock can be shared (another can read the record while an update is in progress) or exclusive (no one can read the record while an update is in progress).

135. Explain issues for database performance.

The installation of the database is an important issue since the better the installation, the better the performance could be. Memory usage is learning how the DBMS uses main memory and then using that knowledge to enable better performance. I/O is usually very intense in a DB, so understanding how users will use the data will help to prepare the database better. CPU usage and application tuning are also important considerations.

136. Describe the difference between homogeneous and heterogeneous distributed database.

A homogenous database is one that uses the same DBMS at each node. Either each node can work independently or a central DBMS may coordinate database activities. A heterogeneous database is one that may have a different DBMS at each node. It may support some or all of the functionality of one logical database. It may support full Distributed DBMS functionality or partial Distributed DBMS functionality.

137. What is a distributed database?

A distributed database is a single logical database that is spread across more than one node or locations that are all connected via some communication link. It requires multiple DBMSs, running at each remote site. A distributed database can be either homogenous (same DBMS at each node) or heterogeneous (different DBMS at some nodes).

138. What is the difference between horizontal and vertical partitioning?

Horizontal partitioning is where some rows of a table are placed into the base relations at one site and other rows are placed at another site. Vertical partitioning is where some columns of a table are placed into the base relations at one site and other columns are placed at another site but each all of these relations must share a common domain.

139. Explain concurrency transparency.

Concurrency transparency is where each transaction in a distributed database is treated as if it is the only one in the system. Therefore if several transactions are running at one time, the results will be the same as if each

transaction was run in serial order. The transaction manager helps to provide concurrency control. The three methods that may be used are locking, versioning, and timestamping.

140. Explain snapshot replication.

Snapshot replication can be used when an application does not require that the data always be current. These applications can be updated by periodic table copying or snapshots. As part of the snapshot effort, all of the data to be included in the snapshot is collected at a primary point. Then a read-only snapshot is taken and the snapshot is sent to each site so that the update can be made.