1. What is SQL?
   1. Structured Query Language
2. What is a relational database management system?
   1. RDBMS. It is a database management system in which the database is organized and accessed according to the relationships between data items. Expressed with tables
3. What is a database?
   1. An organized collection of structured information or data, stored and accessed from a computer system
4. What are the sublanguages of SQL?
   1. Data Definition Language (DDL)
   2. Data Manipulation Language (DML)
   3. Transaction Control Language (TCL)
   4. Data Control Language (DCL)
   5. Data Query Language (DQL)
5. What is cardinality?
   1. > <https://en.wikipedia.org/wiki/Cardinality_(data_modeling)>
   2. Numerical relationship between rows of one table and rows in the other
   3. One-to-one, one-to-many, many-to-many
6. What is a candidate key?
   1. Unique key to identify a record uniquely in a table
7. What is referential integrity?
   1. The logical dependency of a foreign key on a primary key
8. What are primary keys? Foreign keys?
   1. Primary keys are candidate keys that are used to relate one column in a table to the rest. Foreign keys are used to relate rows of one table to rows of another table
9. What are some of the different constraints on columns?
   1. NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY, DEFAULT, CREATE INDEX
10. What is an entity relation diagram?
    1. ERD. Shows the relationships of entity sets stored in a database
11. What are the differences between WHERE vs HAVING?
    1. HAVING can work on aggregated data. Good practice to use WHERE before GROUP BY and HAVING after GROUP BY
12. What are the differences between GROUP BY and ORDER BY?
    1. GROUP BY groups rows that have the same value while ORDER BY sorts
13. What does LIKE do?
    1. LIKE is used to search for a specific pattern in a column
14. How do I use sub queries?
    1. Use it as a nested query in a WHERE clause, SELECT clause, or FROM clause. Use parentheses
15. How does BETWEEN work?
    1. Selects the values in a given range, inclusive on both ends
16. What is the order of operations in an SQL statement?
    1. The order is SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY
17. What is the difference between an aggregate function and a scalar function?
    1. Aggregate functions return a single summarizing value while scalar functions return a value based on scalar inputs
18. What are examples of aggregate and scalar functions?
    1. Aggregate: AVG, SUM, COUNT, MAX, MIN…
    2. Scalar: ROUND, FORMAT, UCASE, LCASE
19. What are the different joins in SQL?
    1. > We have CROSS, INNER, OUTER LEFT, OUTER RIGHT, and OUTER FULL joins. INNER joins only include records with a match in the output (so records where the join condition is true). OUTER joins includes records with a match \*and\* all unmatched records from the left, right, or both tables.
    2. > The part of the JOIN ```ON album.artist\_id = artist.artist\_id``` is the join condition. When the join condition is true for a pair of records, those records are matched together in the output. 90+% of the time, the join condition will be equality based on a foreign key relationship, but you can have various strange join conditions. A join condition of just TRUE will include all pairs of records in the output and is called a CROSS JOIN.
20. What are the different set operations in SQL? Which set operations support duplicates?
    1. > UNION, INTERSECT, UNION ALL are good to know. UNION combines two resultsets removing duplicates, INTERSECT produces results that appear in both of two result sets, and UNION ALL combines two resultsets including duplicates.
21. What is the difference between joins and set operations?
    1. Joins are used to combine columns from different tables while set operations are used to combine rows
22. How can I create an alias in SQL?
    1. Keyword AS
23. What does the AS keyword do in a query?
    1. Creates an alias
24. What is a transaction?
    1. A single unit of work (a single statement)
25. What are the properties of a transaction?
    1. ACID (Atomicity, Consistency, Isolation, Durability)
26. What are the transaction isolation levels and what do they prevent?
    1. Transaction Isolation Levels are a measure of the extent to which transaction isolation succeeds. Levels are defined as the presence or absence of these:
       1. Dirty Reads
       2. Nonrepeatable Reads
       3. Phantoms
    2. The levels are :
       1. Read uncommitted
       2. Read committed
       3. Repeatable read
       4. Serializable
27. What are dirty reads, non repeatable reads, and phantom reads?
28. What is normalization?
    1. The process of organizing the columns and tables of a database to ensure dependencies are properly enforced by database integrity constraints
       1. 1NF
          1. Each table cell should contain a single value
          2. Each record is unique
       2. 2NF
          1. Be in 1NF
          2. Single column primary key that does not functionally depend on any subset of candidate key relation
       3. 3NF
          1. Be in 2NF
          2. Have no transitive functional dependencies
29. What are the requirements for different levels of normalization?
    1. ^^^
30. What is a view?
    1. A view is like a virtual table that contains data from one or more tables. It does not physically exist in the database
31. What is a dao?
    1. Data access object
32. What is the danger of putting values directly into our queries?
    1. Makes database prone to injection
33. What is multiplicity? Examples of 1-to-1, 1-to-N, N-to-N?
    1. specifies the cardinality or number of instances of an EntityType that can be associated with the instances of another EntityType
34. What is an Index?
    1. A quick lookup table that is used for finding records quickly in a database
35. What advantages does creating an Index give us? Disadvantages?
    1. Allows for quicker searching
    2. Decreases performance on inserts updates and deletes
36. What is CRUD?
    1. Create, Read, Update, Delete. The most basic operations that can be performed in SQL
37. What does it mean that an operation or transaction on a data store is atomic?
    1. It means that the entire transaction happens, or it doesn’t happen at all
38. What does ACID stand for?
    1. Atomicity – All or nothing when it comes to completion of a transaction
    2. Consistency – Guarantees that a transaction never leaves your database in a half-finished state
    3. Isolation – Keeps transactions separated until they are finished (one transaction can’t interact with another)
    4. Durability – Database will keep track of pending changed in such a way that the server can recover from an abnormal termination