|  |  |
| --- | --- |
| **## SQL Study Guide** | |
| **### SQL** |  |
| - What is SQL | Structured Query Language |
| - What are the SQL sublanguages; | **DDL – Data Definition Language**  CREATE, ALTER, DROP and TRUNCATE  **DML – Data Manipulation Language**  INSERT, (SELECT\*), UPDATE and DELETE  **DQL\* - Data Query Language**  SELECT  **DCL – Data Control Language**  GRANT and REVOKE  **TCL – Transaction Control Language**  COMMIT, ROLLBACK, SAVEPOINT, RELEASE SET  TRANSACTION. |
| - What are the Key Statements for the Sub Languages |  |
| - What is multiplicity | **Multiplicity** establishes the bounds for data. |
| - What is cardinality | **Cardinality** describes the actual relationship for data. |
| - What is a Primary Key | A **primary key** is a column of table which uniquely identifies each tuple (row) in that table. Primary key enforces integrity constraints to the table. Only one primary key is allowed to use in a table. The primary key does not accept the any duplicate and NULL values. The primary key value in a table changes very rarely so it is chosen with care where the changes can occur in a seldom manner. A primary key of one table can be referenced by foreign key of another table. |
| - What is a Foreign Key | A **foreign key** is a column or group of columns in a relational database table that provides a link between data in two tables. It is a column (or columns) that references a column (most often the primary key) of another table. |
| - What is referential integrity | is a relational database concept, which states that table relationships must always be consistent. |
| - What are the different constraints |  |
| - What do you call a record with a foreign key relationship that doesn’t exist |  |
| - What are the differences between WHERE vs HAVING | The **WHERE clause** works on row’s data, not on aggregated data.    The **HAVING** clause works on aggregated data. |
| - what are the differences between GROUP BY and ORDER BY | **GROUP BY statement** is used to group the rows that have the same value. It is often used with aggregate functions.  **ORDER BY keyword** sort the result-set either in ascending or in descending order. |
| - What is the difference between an aggregate function and a scalar function? | **Aggregate function** - Functions that are used to do operations from the values of the column and a single value is returned.  **Scalar Function** - Functions that are based on user input that return a single value each time it’s invoked. |
| - Name some scalar functions and their usage | UCASE() - It converts the value of a field to uppercase  LCASE() - It converts the value of a field to uppercase  MID() - extracts texts from the text field.  LEN() - returns the length of the value in a text field.  ROUND() - used to round a numeric field to the number of decimals specified.  NOW() - the current system date and time  FORMAT() - used to format how a field is to be displayed. |
| - Name some aggregate functions and their usage | AVG() - returns average value after calculating from values in a numeric column  COUNT() - used to count the number of rows returned in a SELECT statement.  FIRST() - returns the first value of the selected column  LAST() - returns the last value of the selected column.  MAX() - returns the maximum value of the selected column.  MIN() - returns the minimum value of the selected column  SUM() - returns the sum of all the values of the selected column. |
| - What does LIKE do? | It is useful when you want to search rows to match a specific pattern, or when you do not know the entire value. |
| - How do I use sub queries? |  |
| - How does BETWEEN work? | The **BETWEEN** and **AND** operators are used in WHERE clauses to compare data for a range of values. |
| - What is the order of operations in an SQL statement? | 1. FROM 2. WHERE 3. GROUP BY 4. HAVING 5. SELECT 6. ORDER BY |
| - What are the different joins in SQL? |  |
| - What are the different set operations in SQL? |  |
| - Difference between union and union all |  |
| - What is the difference between joins and set operations? |  |
| - How can I create a alias in SQL | as alias\_name |
| - What does the AS keyword do in a query? | Aliases are the temporary names given to table or column for the purpose of a particular SQL query. It is used when name of column or table is used other than their original names, but the modified name is only temporary. |
| - What are the properties of a transaction? |  |
| - What are the transaction isolation levels and what do they prevent? |  |
| - What are dirty reads, non repeatable reads, and phantom reads? |  |
| - What is normalization | Normalization is the method used in a database to reduce the data redundancy and data inconsistency from the table. It is the technique in which Non-redundancy and consistency data are stored in the set schema. By using normalization the number of tables is increased instead of decreased. |
| - What are the requirements for the different normalization levels |  |
| - What is pl/pgsql |  |
| - What are triggers |  |
|  |  |
| **### MAVEN** | |
| - What is Maven? |  |
| - What are the Maven Lifecycles? |  |
| - What is the purpose of the POM.xml? |  |
|  |  |
| **### Design Patterns** | |
| - What is the difference between Singleton and Factory? |  |
|  |  |
| **### JDBC** |  |
| - What is JDBC |  |
| - What are the interfaces and classes of JDBC |  |
| - What are the different types of statements |  |
| - What is SQL Injection |  |
| - What is a DAO |  |
|  |  |
|  |  |