50 Most Common SQL Questions and Answers

Basic Queries

1. **What is SQL?**
   - SQL (Structured Query Language) is a standardized programming language used to manage relational databases and perform various operations on the data in them.

2. **What are the different types of SQL commands?**
   - DDL (Data Definition Language): CREATE, ALTER, DROP, TRUNCATE
   - DML (Data Manipulation Language): SELECT, INSERT, UPDATE, DELETE
   - DCL (Data Control Language): GRANT, REVOKE
   - TCL (Transaction Control Language): COMMIT, ROLLBACK, SAVEPOINT

3. **How do you select all columns from a table?**

   ```
   SELECT * FROM table_name;
   ```

4. **How do you select specific columns from a table?**

   ```
   SELECT column1, column2 FROM table_name;
   ```

5. **What is the WHERE clause used for?**
   - The WHERE clause is used to filter records based on specified conditions.

   ```
   SELECT * FROM table_name WHERE condition;
   ```

6. **How do you sort results in SQL?**

   ```
   SELECT column1 FROM table_name ORDER BY column1 ASC/DESC;
   ```

Joins and Relationships

7. **What are the different types of joins in SQL?**
   - INNER JOIN: Returns matching records from both tables
   - LEFT JOIN: Returns all records from left table and matching from right
   - RIGHT JOIN: Returns all records from right table and matching from left
   - FULL JOIN: Returns all records when there's a match in either left or right table

8. **How do you write an INNER JOIN?**

   ```
   SELECT * FROM table1
   INNER JOIN table2
   ON table1.column = table2.column;
   ```

9. **What's the difference between INNER and LEFT JOIN?**
   - INNER JOIN returns only matching records
   - LEFT JOIN returns all records from left table and matching from right

10. **How do you join multiple tables?**

    ```
    SELECT * FROM table1
    JOIN table2 ON table1.id = table2.id
    JOIN table3 ON table2.id = table3.id;
    ```

Aggregate Functions

11. **What are aggregate functions in SQL?**
    - Functions that perform calculations on a set of values: COUNT, SUM, AVG, MAX, MIN

12. **How do you count records in a table?**

    ```
    SELECT COUNT(*) FROM table_name;
    ```

13. **How do you find the sum of a column?**

    ```
    SELECT SUM(column_name) FROM table_name;
    ```

14. **What is GROUP BY used for?**
    - Groups rows that have the same values in specified columns

```
SELECT column1, COUNT(*) FROM table_name
GROUP BY column1;
```

Conditions and Filtering

15. **What is the difference between WHERE and HAVING?**
    - WHERE filters individual rows before grouping
    - HAVING filters groups after GROUP BY

16. **How do you use the IN operator?**

```
SELECT * FROM table_name
WHERE column_name IN ('value1', 'value2');
```

17. **How do you use the BETWEEN operator?**

```
SELECT * FROM table_name
WHERE column_name BETWEEN value1 AND value2;
```

18. **What is the LIKE operator used for?**
    - Pattern matching in strings using wildcards (% and _)

```
SELECT * FROM table_name
WHERE column_name LIKE 'pattern%';
```

Database Management

19. **How do you create a new table?**

```
CREATE TABLE table_name (
    column1 datatype,
    column2 datatype,
    PRIMARY KEY (column1)
);
```

20. **How do you add a new column to an existing table?**

```
ALTER TABLE table_name
ADD column_name datatype;
```

21. **How do you delete a table?**

```
DROP TABLE table_name;
```

22. **What's the difference between DROP and TRUNCATE?**
    - DROP deletes the table structure and data
    - TRUNCATE deletes all data but keeps the table structure

Data Manipulation

23. **How do you insert data into a table?**

```
INSERT INTO table_name (column1, column2)
VALUES (value1, value2);
```

24. **How do you update existing records?**

```
UPDATE table_name
SET column1 = value1
WHERE condition;
```

25. **How do you delete records?**

```
DELETE FROM table_name
WHERE condition;
```

Advanced Concepts

26. **What is a subquery?**

    - A query nested inside another query

    ```
    SELECT * FROM table1
    WHERE column1 IN (SELECT column1 FROM table2);
    ```

27. **What are views?**

    - Virtual tables based on the result set of an SQL statement

    ```
    CREATE VIEW view_name AS
    SELECT column1, column2
    FROM table_name;
    ```

28. **What is an index?**

    - Database structure that improves the speed of data retrieval

    ```
    CREATE INDEX index_name
    ON table_name (column1, column2);
    ```

29. **What are stored procedures?**

    - Prepared SQL code that can be saved and reused

    ```
    CREATE PROCEDURE procedure_name
    AS
    sql_statement
    GO;
    ```

30. **What is a trigger?**

    - Special stored procedure that automatically runs when an event occurs

    ```
    CREATE TRIGGER trigger_name
    ON table_name
    AFTER INSERT
    AS
    BEGIN
        -- trigger logic
    END;
    ```

Data Types and Constraints

31. **What are the common SQL data types?**

    - VARCHAR: Variable-length string
    - INT: Integer numbers
    - DATETIME: Date and time values
    - DECIMAL: Exact numeric values
    - BOOLEAN: True/false values

32. **What is a PRIMARY KEY?**

    - Unique identifier for each record in a table
    - Cannot contain NULL values
    - Must be unique

33. **What is a FOREIGN KEY?**

    - Column that creates a relationship between two tables
    - References PRIMARY KEY of another table

34. **What is a UNIQUE constraint?**

    - Ensures all values in a column are different

Performance and Optimization

35. **How can you optimize SQL queries?**

- Use indexes appropriately
- Avoid SELECT *
- Use appropriate JOIN types
- Limit the use of subqueries
- Use EXPLAIN to analyze query performance

36. **What is query caching?**

- Storing query results in memory for faster retrieval
- Subsequent identical queries can be served from cache

37. **How do you handle NULL values?**

```
 SELECT * FROM table_name
WHERE column_name IS NULL;
-- or
WHERE column_name IS NOT NULL;
```

Common Operations

38. **How do you find duplicate records?**

```
 SELECT column1, COUNT(*)
FROM table_name
GROUP BY column1
HAVING COUNT(*) > 1;
```

39. **How do you use CASE statements?**

```
 SELECT column1,
CASE
    WHEN condition1 THEN result1
    WHEN condition2 THEN result2
    ELSE result3
END
FROM table_name;
```

40. **How do you combine results from multiple queries?**

```
 SELECT column1 FROM table1
UNION
SELECT column1 FROM table2;
```

String and Date Functions

41. **How do you concatenate strings?**

```
 SELECT CONCAT(column1, ' ', column2)
FROM table_name;
```

42. **How do you extract parts of dates?**

```
 SELECT YEAR(date_column)
FROM table_name;
```

Window Functions

43. **What are window functions?**

- Functions that perform calculations across rows related to current row

```
 SELECT column1,
ROW_NUMBER() OVER (ORDER BY column1) as row_num
FROM table_name;
```

44. **How do you use PARTITION BY?**

```
 SELECT column1,
AVG(column2) OVER (PARTITION BY column1)
FROM table_name;
```

Transaction Management

45. **What is a transaction?**

    - A unit of work that must be completed in its entirety

    ```
    BEGIN TRANSACTION;
        -- SQL statements
    COMMIT;
    ```

46. **What are the ACID properties?**

    - Atomicity: Transaction is all or nothing
    - Consistency: Database remains consistent
    - Isolation: Transactions are independent
    - Durability: Changes are permanent

Security

47. **How do you grant permissions?**

    ```
    GRANT SELECT, INSERT ON table_name
    TO user_name;
    ```

48. **How do you revoke permissions?**

    ```
    REVOKE SELECT ON table_name
    FROM user_name;
    ```

Error Handling

49. **How do you handle errors in SQL?**

    ```
    BEGIN TRY
        -- SQL statements
    END TRY
    BEGIN CATCH
        -- Error handling
    END CATCH
    ```

50. **What is SQL injection and how to prevent it?**

    - Security vulnerability where malicious SQL code is inserted
    - Prevention:
        - Use parameterized queries
        - Input validation
        - Escape special characters
        - Use stored procedures