Question-1

In SQL, a NULL value represents the absence of any data in a field. It essentially indicates an unknown or missing value rather than zero or an empty string. NULL is a distinct entity that is used when data is unavailable or inapplica-ble for a specific field

1. NULL ≠ 0:

Zero is a specific number, meaning "nothing" or "none" quantitatively. In contrast, NULL has no quantitative meaning, it means that the value is unknown or undefined

2. NULL ≠ Empty String:

An empty string ('') is a known value that simply contains no characters, while NULL indicates an absence of data altogether

3. NULL as Unique:

NULL is treated as a unique state that is not equal to any other value, including another NULL

Question - 2

* Where Clauses →

When a NULL appears in a WHERE clause, it requires explicit handling since comparisons with NULL do not return true or false but rather "unknown". To check for NULL values, SQL uses the IS NULL and IS NOT NULL keywords instead of traditional equality operators (= or !=)

* JOIN Conditions →

In joins, NULL values can affect results since they are not considered equal to any other value, including another NULL. As a result, if a JOIN condition relies on matching NULL values b/w tables, it will exclude rows where NULLs are present unless specifically handled with IS NULL conditions.

* Arithmetic Operations →

Arithmetic operations involving NULL yield NULL results, as any operation with an unknown value cannot produce a definitive outcome for instance, 10 + NULL results in NULL.

Question - 3

NULL values interact with aggregate functions in SQL differently →

1. SUM

Ignores NULL values, calculating the sum of only non-NULL values.

2. COUNT

COUNT(*) includes NULLs because it counts all rows but COUNT(column_name) ignores NULLs, counting only non-NULL entries in the specified column

3. AVG

Calculates the average of only non-NULL values, excluding NULLs from both the sum and the count

4. MIN and MAX

NULLs are ignored when determining the minimum or maximum values in a column

Thus NULLs are effectively "invisible" in aggregate calculations, meaning that the presence of NULLs do not affect the outcome unless all values in the column are NULL, in which case some functions return NULL (eg. SUM, AVG) as there's no data to operate on.

Question-4

Most SQL aggregate functions ignore NULLs by default →

1. SUM, AVG, MIN, MAX, COUNT(column_name)

these functions exclude NULL values from their calculations, focusing only on rows with non-NULL values.

2. COUNT(*)

unlike others, COUNT(*) includes NULLs because it counts every row regardless of column values

Question-5

```
CREATE TABLE Sales ( Salesperson_ID INT,
                     SalesAmount DECIMAL(10,2)
                   );

INSERT INTO Sales VALUES
  (1, 500.00),
  (2, NULL),
  (3, 600.85),
  (4, NULL),
  (5, 700.00);

SELECT SUM(SalesAmount) AS TotalSales,
       COUNT(SalesAmount) AS NonNULLSales,
       COUNT(*) AS TotalRows
FROM Sales;
```