**1. Date Functions**

These functions are used to manipulate and extract information from date fields.

* **DATE()** – Converts a date or datetime string to a date type.
* **DATEADD()** – Adds a specified number of time intervals to a date.
* **DATEDIFF()** – Returns the difference between two dates in a specified date part (e.g., days, months, years).
* **DATENAME()** – Returns the name of the date part (e.g., "January", "Monday").
* **DATEPART()** – Returns the numeric value of a date part (e.g., 1 for January, 3 for Wednesday).
* **DATETRUNC()** – Truncates a date to a specified date part (e.g., year, quarter, month).
* **TODAY()** – Returns the current date (no time).
* **NOW()** – Returns the current date and time.
* **MAKEDATE()** – Creates a date from year, month, and day values.
* **MAKETIME()** – Creates a time from hour, minute, and second values.
* **DATEPARSE()** – Parses a string into a date based on a specified format.

**2. String Functions**

These functions allow you to manipulate text strings.

* **CONTAINS()** – Checks if a string contains a specified substring.
* **LEFT()** – Returns the left part of a string.
* **LEN()** – Returns the length of a string.
* **LOWER()** – Converts a string to lowercase.
* **UPPER()** – Converts a string to uppercase.
* **MID()** – Extracts a substring from a string.
* **TRIM()** – Removes leading and trailing spaces from a string.
* **REPLACE()** – Replaces occurrences of a substring with another substring.
* **REGEXP\_EXTRACT()** – Extracts a substring from a string using a regular expression.
* **REGEXP\_REPLACE()** – Replaces parts of a string based on a regular expression.
* **REGEXP\_MATCH()** – Checks if a string matches a regular expression.
* **FIND()** – Returns the position of a substring within a string.
* **SPLIT()** – Splits a string into substrings based on a delimiter.
* **TRIM()** – Removes extra spaces from a string.

**3. Number Functions**

These functions are used to perform mathematical and numeric operations.

* **ABS()** – Returns the absolute value of a number.
* **CEILING()** – Rounds a number up to the nearest integer.
* **FLOOR()** – Rounds a number down to the nearest integer.
* **ROUND()** – Rounds a number to a specified number of decimal places.
* **RANK()** – Ranks values in a dataset.
* **SUM()** – Sums up a field.
* **AVG()** – Returns the average of a field.
* **MIN()** – Returns the minimum value of a field.
* **MAX()** – Returns the maximum value of a field.
* **COUNT()** – Counts the number of records in a field.
* **COUNTD()** – Counts the number of distinct values in a field.
* **EXP()** – Returns e raised to the power of the given number.
* **LN()** – Returns the natural logarithm of a number.
* **LOG()** – Returns the logarithm of a number to a specified base.
* **PI()** – Returns the value of Pi.
* **POWER()** – Raises a number to a specified power.
* **SQRT()** – Returns the square root of a number.

**4. Logical Functions**

These functions allow you to perform logical tests and return different results.

* **IF / IFNULL** – Conditional statements for testing logical conditions.
* **CASE / WHEN** – Similar to IF but in a cleaner syntax for multiple conditions.
* **ISNULL()** – Checks whether a value is null.
* **IFNULL()** – Returns a specified value if the input is null.
* **NULLIF()** – Returns null if two values are equal, otherwise returns the first value.
* **ZNULL()** – Creates a null value.
* **AND, OR, NOT** – Logical operators for combining conditions.

**5. Aggregate Functions**

These functions perform calculations across multiple rows of data.

* **SUM()** – Sums up values in a field.
* **AVG()** – Returns the average of values in a field.
* **MIN()** – Returns the minimum value from a group of values.
* **MAX()** – Returns the maximum value from a group of values.
* **COUNT()** – Counts the number of rows in a dataset.
* **COUNTD()** – Counts the number of distinct values in a field.
* **STDEV()** – Returns the standard deviation of values.
* **STDEVP()** – Returns the standard deviation for a population.
* **VAR()** – Returns the variance of values.
* **VARP()** – Returns the variance for a population.

**6. Type Conversion Functions**

These functions are used to convert one data type to another.

* **INT()** – Converts a value to an integer.
* **FLOAT()** – Converts a value to a floating-point number.
* **STR()** – Converts a value to a string.
* **DATE()** – Converts a string or numeric value to a date.
* **DATETIME()** – Converts a string or numeric value to a datetime.
* **BOOLEAN()** – Converts a value to a boolean (true or false).

**7. Table Calculation Functions**

These functions are used to perform calculations based on the context of a visualization.

* **RUNNING\_SUM()** – Calculates the cumulative sum of a field.
* **RUNNING\_AVG()** – Calculates the cumulative average of a field.
* **WINDOW\_SUM()** – Sums a field within a window of data.
* **WINDOW\_AVG()** – Averages a field within a window of data.
* **LOOKUP()** – Returns the value of a field from a previous or next row.
* **INDEX()** – Returns the index of a row in the partition.
* **FIRST()** – Returns the index of the first row in a partition.
* **LAST()** – Returns the index of the last row in a partition.
* **PREVIOUS\_VALUE()** – Returns the previous value of a field in a calculation.

**8. Window Functions**

Window functions are used to perform computations across a set of rows related to the current row.

* **WINDOW\_AVG()** – Returns the average of a field within the window of data.
* **WINDOW\_SUM()** – Returns the sum of a field within the window of data.
* **WINDOW\_MIN()** – Returns the minimum value within the window of data.
* **WINDOW\_MAX()** – Returns the maximum value within the window of data.

**9. Miscellaneous Functions**

These are a few additional functions that don’t fall into the other categories but are still useful.

* **ISNULL()** – Returns true if a value is null.
* **ISDATE()** – Returns true if a value is a valid date.
* **ISNUMBER()** – Returns true if a value is a number.
* **ISBLANK()** – Returns true if a value is blank (null or empty).
* **RANDOM()** – Returns a random number between 0 and 1.
* **UUID()** – Returns a universally unique identifier (UUID).

**10. Data Source Functions**

These functions are used to manipulate data within the data source or database.

* **EXTRACT()** – Extracts a subset of data from a data source (for use in Tableau Data Extracts).
* **JOIN()** – Specifies how two or more data sources should be joined.
* **UNION()** – Combines data from two or more tables with the same structure.