**🧠 1. Basic String Functions**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| LENGTH() / LEN() | Returns length of a string | SELECT LENGTH('Hello'); | 5 |
| UPPER() / UCASE() | Converts to uppercase | SELECT UPPER('hello'); | HELLO |
| LOWER() / LCASE() | Converts to lowercase | SELECT LOWER('HELLO'); | hello |
| INITCAP() *(Oracle, PostgreSQL)* | Capitalizes first letter of each word | SELECT INITCAP('sql string functions'); | Sql String Functions |
| REVERSE() | Reverses string | SELECT REVERSE('SQL'); | LQS |

**✂️ 2. Trimming and Padding**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| TRIM() | Removes spaces from both ends | SELECT TRIM(' Hello '); | Hello |
| LTRIM() | Removes leading spaces | SELECT LTRIM(' Hello'); | Hello |
| RTRIM() | Removes trailing spaces | SELECT RTRIM('Hello '); | Hello |
| LPAD(str, len, padstr) | Pads string on left | SELECT LPAD('SQL', 6, '\*'); | \*\*\*SQL |
| RPAD(str, len, padstr) | Pads string on right | SELECT RPAD('SQL', 6, '\*'); | SQL\*\*\* |

**🔍 3. Searching and Extracting**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| SUBSTRING(str, start, length) / SUBSTR() | Extracts substring | SELECT SUBSTRING('Database', 5, 3); | bas |
| LEFT(str, n) | Returns first *n* chars | SELECT LEFT('Database', 4); | Data |
| RIGHT(str, n) | Returns last *n* chars | SELECT RIGHT('Database', 4); | base |
| INSTR(str, substr) / CHARINDEX() | Finds position of substring | SELECT INSTR('Database', 'base'); | 5 |
| LOCATE(substr, str) | Same as INSTR | SELECT LOCATE('a', 'Database'); | 2 |

**🔄 4. Modifying Strings**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| CONCAT() | Joins strings | SELECT CONCAT('SQL', ' ', 'Server'); | SQL Server |
| CONCAT\_WS(separator, str1, str2, …) | Joins with separator | SELECT CONCAT\_WS('-', 'SQL', '2025'); | SQL-2025 |
| REPLACE(str, from\_str, to\_str) | Replaces substring | SELECT REPLACE('SQL Tutorial', 'SQL', 'MySQL'); | MySQL Tutorial |
| REPEAT(str, count) | Repeats string | SELECT REPEAT('Hi', 3); | HiHiHi |

**🧩 5. Comparison and Formatting**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| STRCMP(str1, str2) | Compares two strings (returns 0 if equal) | SELECT STRCMP('abc', 'abc'); | 0 |
| FIELD(str, str1, str2, …) | Returns index of matching value | SELECT FIELD('b', 'a', 'b', 'c'); | 2 |
| FORMAT(number, decimal\_places) | Formats number as string | SELECT FORMAT(1234.567, 2); | 1,234.57 |

**⚙️ 6. Advanced / Special String Functions**

| **Function** | **Description** | **Example** | **Output** |
| --- | --- | --- | --- |
| QUOTE(str) | Adds quotes around a string | SELECT QUOTE('Hello'); | 'Hello' |
| SPACE(n) | Returns n spaces | SELECT CONCAT('Hi', SPACE(3), 'SQL'); | Hi SQL |
| ASCII(char) | Returns ASCII code | SELECT ASCII('A'); | 65 |
| CHAR(code) | Returns character for ASCII code | SELECT CHAR(65); | A |
| REPLACE(REPLACE(str,'old','new'),'x','y') | Nested replace | SELECT REPLACE(REPLACE('abc','a','x'),'b','y'); | xyc |

**🔠 7. Practical Examples**

**✅ Example 1: Extract domain from email**

SELECT SUBSTRING(Email, INSTR(Email, '@') + 1) AS Domain

FROM Employees;

**✅ Example 2: Format names (capitalize)**

SELECT CONCAT(UPPER(LEFT(Name,1)), LOWER(SUBSTRING(Name,2))) AS ProperName

FROM Students;

**✅ Example 3: Mask phone number**

SELECT CONCAT('XXXXXXX', RIGHT(Phone, 3)) AS MaskedPhone

FROM Contacts;

**💡 Notes by SQL Dialect**

| **DBMS** | **Key Differences** |
| --- | --- |
| **MySQL** | Uses LENGTH(), LOCATE(), SUBSTRING() |
| **SQL Server** | Uses LEN(), CHARINDEX(), SUBSTRING() |
| **Oracle** | Uses LENGTH(), INSTR(), SUBSTR(), INITCAP() |
| **PostgreSQL** | Similar to Oracle; supports regex functions too |