hex. For more information about that option, see Section 4.5.1, "mysql — The MySQL Command-Line Client".

Data Masking Functions

Each function in this section performs a masking operation on its string argument and returns the masked result.

- mask_inner(str, margin1, margin2 [, mask_char])

  Masks the interior part of a string, leaving the ends untouched, and returns the result. An optional masking character can be specified.

  Arguments:

  - str: The string to mask.

  - margin1: A nonnegative integer that specifies the number of characters on the left end of the string to remain unmasked. If the value is 0, no left end characters remain unmasked.

  - margin2: A nonnegative integer that specifies the number of characters on the right end of the string to remain unmasked. If the value is 0, no right end characters remain unmasked.

  - mask_char: (Optional) The single character to use for masking. The default is 'X' if mask_char is not given.

    The masking character must be a single-byte character. Attempts to use a multibyte character produce an error.

  Return value:

  The masked string, or NULL if either margin is negative.

  If the sum of the margin values is larger than the argument length, no masking occurs and the argument is returned unchanged.

  Example:

  ```
  mysql> SELECT mask_inner('abcdef', 1, 2), mask_inner('abcdef',0, 5);
  +--------------------------+--------------------------+
  | mask_inner('abcdef', 1, 2) | mask_inner('abcdef',0, 5) |
  +--------------------------+--------------------------+
  | aXXXef                   | Xbcdef                   |
  +--------------------------+--------------------------+
  mysql> SELECT mask_inner('abcdef', 1, 2, '*'), mask_inner('abcdef',0, 5, '#');
  +-------------------------------+-------------------------------+
  | mask_inner('abcdef', 1, 2, '*') | mask_inner('abcdef',0, 5, '#') |
  +-------------------------------+-------------------------------+
  | a***ef                        | #bcdef                        |
  +-------------------------------+-------------------------------+
  ```

- mask_outer(str, margin1, margin2 [, mask_char])

  Masks the left and right ends of a string, leaving the interior unmasked, and returns the result. An optional masking character can be specified.

  Arguments:

  - str: The string to mask.