

*/*Introducing SP_ESCAPE and SP_UNESCAPE stored procedures.*

Fabiano Bonin

*Want to put text with line feeds, carriage returns, quotes and tabs in
you sql scripts?*

```
select result from SP_UNESCAPE('word1\tword2\r\n\\line2\\r\n\quoted  
text\q')
```

The procedure above will return:

word1word2

\line2

'quoted text'

\t - will be replaced for a tab

\n - will be replaced for a line feed

\r - will be replaced for a carriage return

\q - will be replaced for a quote

\\ - will be replaced for a backslash

*Want to do the opposite - convert the value of a text field to a
escaped sequence?*

```
select result from SP_ESCAPE(your_text_field)
```

The procedure above will return the text in the field escaped

following the same rules above (inverted)

*Works with char, varchar, and text blob fields.**

```
SET TERM ^ ;
CREATE OR ALTER PROCEDURE sp_escape (
    text BLOB SUB_TYPE 1
)
RETURNS (
    result BLOB SUB_TYPE 1
)
AS
    DECLARE VARIABLE i INTEGER;
    DECLARE VARIABLE l INTEGER;
    DECLARE VARIABLE c VARCHAR(1);
    DECLARE VARIABLE e VARCHAR(2);
BEGIN
    IF (
        text IS NULL) THEN
    BEGIN
        result = NULL;
        SUSPEND;
        EXIT;
    END

    i = 1;
    l = char_length(text);
    result = '';
    WHILE (
        i <=
        l) DO
    BEGIN
        c = substring(text FROM
            i
        FOR
            1);
        IF (
            c =
            '\') THEN
            e = '\\';
        ELSE
            IF (
                c =
                '''') THEN
                    e = '\q';
                ELSE
                    IF (
                        ascii_val(c) =
                        9) THEN
                            e = '\t';
                        ELSE
                            IF (
                                ascii_val(c) =
                                10) THEN
                                    e = '\n';
```

```

        ELSE
            IF (
                ascii_val(c) =
                13) THEN
                e = '\r';
            ELSE
                e = c;
            result =
                result ||
                e;
            i =
                i +
                1;
        END
        SUSPEND;
    END ^
    SET TERM ; ^

```

```

SET TERM ^ ;
CREATE OR ALTER PROCEDURE sp_unescape (
    texto BLOB SUB_TYPE 1
)
RETURNS (
    result BLOB SUB_TYPE 1
)
AS
    DECLARE VARIABLE i INTEGER;
    DECLARE VARIABLE l INTEGER;
    DECLARE VARIABLE c VARCHAR(2);
    DECLARE VARIABLE e VARCHAR(2);
BEGIN
    IF (
        texto IS NULL) THEN
        BEGIN
            result = NULL;
            SUSPEND;
            EXIT;
        END

        i = 1;
        l = char_length(texto);
        result = '';
        WHILE (
            i <=
            l) DO
            BEGIN
                c = substring(texto FROM
                    i
                    FOR
                    2);
                IF (
                    c =
                    '\\') THEN
                BEGIN
                    e = '\\';

```

```

        i =
            i +
            1;
    END ELSE
    IF (
        c =
            '\q') THEN
    BEGIN
        e = ''';
        i =
            i +
            1;
    END ELSE
    IF (
        c =
            '\t') THEN
    BEGIN
        e = ascii_char(9);
        i =
            i +
            1;
    END ELSE
    IF (
        c =
            '\n') THEN
    BEGIN
        e = ascii_char(10);
        i =
            i +
            1;
    END ELSE
    IF (
        c =
            '\r') THEN
    BEGIN
        e = ascii_char(13);
        i =
            i +
            1;
    END ELSE
        e = substring(c FROM 1 FOR 1);
    result =
        result ||
        e;
    i =
        i +
        1;
END
SUSPEND;
END;
SET TERM ;

```