



pears.GetTextFromHTML



Generated schema reference. Regenerate this page from the SQL unload; keep hand-maintained business notes in the narrative namespace.

Original SQL

```
COMMENT TO PRESERVE FORMAT ON PROCEDURE "pears"."GetTextFromHTML" IS
{CREATE FUNCTION GetTextFromHTML

/* Application Maintained Function / Procedure - DO NOT EDIT*/

( IN htmlText long VARCHAR )
RETURNS long VARCHAR
BEGIN
    DECLARE plainText long VARCHAR;
    DECLARE tagStart INTEGER;
    DECLARE tagEnd INTEGER;
    DECLARE tagLength INTEGER;
    DECLARE tagType CHAR(20);
    DECLARE endOfTagName INTEGER;
    SET plainText = htmlText;
    IF locate( plainText, '<HEAD>', 1 ) = 0 THEN
        RETURN plainText;
    END IF;
    SET plainText = RIGHT(plainText, LENGTH(plainText) -
charindex('</HEAD>', UPPER(plainText)) - 6);
    SET plainText = REPLACE(plainText, CHAR(13), '');
    SET plainText = REPLACE(plainText, CHAR(10), '');
    SET plainText = REPLACE(plainText, ',', CHAR(96));
    SET plainText = REPLACE(plainText, '@', CHAR(96));
    while locate( plainText, '<', 1 ) > 0 loop
        SET tagStart = charindex('<', plainText);
        SET tagEnd = charindex('>', plainText);
        // make tag TYPE always be IN UPPER CASE, AND compensate FOR the <
things
        SET tagType = SUBSTRING(UPPER(plainText), tagStart+1, tagEnd-
tagStart-1);
        SET tagLength = tagEnd - tagStart;
        // CHECK TO see IF there are spaces IN the name (there are
attributes).
        // IF there are GET rid OF them
```



```
SET endOfTagName = charindex(' ', tagType);
IF endOfTagName > 0 THEN
    SET tagType = SUBSTRING(tagType, 1, endOfTagName);
END IF;
CASE tagType
    WHEN '/DIV' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/P' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H1' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H2' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H3' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H4' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H5' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/H6' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/UL' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/OL' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN 'BR' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/TABLE' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN 'HR' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/ARTICLE' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/ASIDE' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/BLOCKQUOTE' THEN SET plainText = stuff(plainText,
tagStart, tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/FIGCAPTION' THEN SET plainText = stuff(plainText,
tagStart, tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/FIGURE' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/FOOTER' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/FORM' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/HEADER' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/MENU' THEN SET plainText = stuff(plainText, tagStart,
```



```
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/NAV' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/PRE' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/SECTION' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    WHEN '/SUMMARY' THEN SET plainText = stuff(plainText, tagStart,
tagLength+1, CHAR(13)+CHAR(10));
    ELSE SET plainText = stuff(plainText, tagStart, tagLength+1,
NULL);
    END;
END loop;
SET plainText = REPLACE(plainText, '&nbsp;',' ');
SET plainText = REPLACE(plainText, '&','&');
SET plainText = REPLACE(plainText, '>','>');
SET plainText = REPLACE(plainText, '<','<');
SET plainText = REPLACE(plainText, '"','CHAR(34)');
SET plainText = REPLACE(plainText, ''','CHAR(39)');
SET plainText = REPLACE(plainText, '¢','CHAR(163)');
SET plainText = REPLACE(plainText, '£','CHAR(163)');
SET plainText = REPLACE(plainText, '€','CHAR(128)');
SET plainText = REPLACE(plainText, '¥','CHAR(165)');
SET plainText = REPLACE(plainText, '©','CHAR(169)');
SET plainText = REPLACE(plainText, '®','CHAR(174)');
IF (isnull(plainText,'') = '') AND (isnull(htmltext,'')<>'') THEN
    SET plainText = 'Could not parse following HTML:' +
CHAR(13)+CHAR(10) + htmlText;
endif;
RETURN plainText;
END
}
```

From:

<https://iqxusers.co.uk/iqxhelp/> - iqx

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears_gettextfromhtml

Last update: 2026/08/07 19:24

