



pears.GetQuestTextLong



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"."GetQuestTextLong" IS
{CREATE FUNCTION GetQuestTextLong

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

(IN tagloc CHAR(3),IN wilddtagid CHAR(3),IN parentid CHAR(20))
RETURNS long VARCHAR
BEGIN
  DECLARE rv long VARCHAR;
  DECLARE tagval DOUBLE;
  DECLARE textval long VARCHAR;
  DECLARE x CHAR(4);
  DECLARE y INTEGER;
  DECLARE xunits CHAR(11);
  DECLARE xdesc CHAR(25);
  DECLARE i SMALLINT;
  DECLARE tab CHAR(1);
  DECLARE newline CHAR(2);
  SET tab="char"(9);
  SET newline="char"(13)+"char"(10);
  SET rv='';
  FOR fetchfor AS fetchcursor no scroll cursor FOR
    SELECT tagid AS toptagid,(IF isnull(longdescription,'')='' THEN
description ELSE longdescription endif) AS tagdesc,tagtype,units FROM tag
WHERE taglocation = tagloc AND tagid LIKE wilddtagid ORDER BY sortorder ASC
FOR READ ONLY do
  SET xunits=isnull(units,'');
  IF xunits <> '' THEN
    SET xunits=' '+xunits
  END IF;
  SET rv=rv+tagdesc+tab;
  CASE tagtype WHEN 'N' THEN
    SELECT MAX(VALUE) INTO tagval FROM tagvalue WHERE taglocation = tagloc
AND id = parentid AND tagid = toptagid;
    IF isnull(tagval,0) <> 0 THEN
```



```
        SET rv=rv+string(tagval)+xunits
    END IF WHEN 'D' THEN
        SELECT MAX(VALUE) INTO tagval FROM tagvalue WHERE taglocation = tagloc
    AND id = parentid AND tagid = toptagid;
    IF isnull(tagval,0) <> 0 THEN
        SET y=tagval; //// Following expression does NOT behave WITH a
    DOUBLE
        SET rv=rv || dateformat("date"('1899-12-30')+y,'dd/mm/yyyy')
    END IF WHEN 'T' THEN
        SELECT MAX(textvalue) INTO textval FROM tagvalue WHERE taglocation =
    tagloc AND id = parentid AND tagid = toptagid;
    IF textval IS NOT NULL THEN
        SET rv=rv+textval
    END IF WHEN 'U' THEN
        SELECT MAX(textvalue) INTO textval FROM tagvalue WHERE taglocation =
    tagloc AND id = parentid AND tagid = toptagid;
    IF textval IS NOT NULL THEN
        SET rv=rv+textval
    END IF WHEN 'M' THEN
        SELECT MAX(tagchoiceid) INTO x FROM tagvalue WHERE taglocation =
    tagloc AND id = parentid AND tagid = toptagid;
    IF x IS NULL THEN
        SET x='_'
    END IF;
    SET rv=rv+isnull((SELECT (IF isnull(longdescription,'')='' THEN
    description ELSE longdescription endif) FROM tagchoice WHERE taglocation =
    tagloc AND tagid = toptagid AND tagchoiceid = x),'')
    ELSE
        SET i=0;
        FOR fetch2for AS fetch2cursor no scroll cursor FOR
            SELECT (IF isnull(tagchoice.longdescription,'')='' THEN
            tagchoice.description ELSE tagchoice.longdescription endif) AS
            tagchdesc,tagvalue.tagchoiceid,tagvalue.value AS tgval,
            tagchoice.tagchoiceparentid AS tcpid
            FROM tagvalue KEY JOIN tagchoice
            WHERE tagvalue.taglocation = tagloc AND tagvalue.tagid = toptagid
    AND tagvalue.id = parentid ORDER BY tagchoice.sortorder ASC FOR READ ONLY do
            IF i > 0 THEN
                SET rv=rv+', '
            END IF;
            CASE tagtype WHEN 'L' THEN
                SET rv=rv+tagchdesc;
                IF tagchoiceid = '*' THEN
                    SET rv=rv+' '+string(tgval)
                END IF WHEN 'S' THEN
                    SET rv=rv+tagchdesc+' '+string(tgval)+xunits WHEN 'G' THEN
                    SELECT MAX((IF isnull(longdescription,'')='' THEN description ELSE
                    longdescription endif)) INTO xdesc FROM tagchoice WHERE taglocation = tagloc
```



```
AND tagid = toptagid AND VALUE = tgval AND subchoice = 1;
    IF isnull(xdesc,'') <> '' THEN
        SET rv=rv+tagchdesc+' '+xdesc
    END IF WHEN 'Q' THEN
        SELECT MAX((IF isnull(longdescription,'')='' THEN description ELSE
longdescription endif)) INTO xdesc FROM tagchoice WHERE taglocation = tagloc
AND tagid = toptagid AND tagchoiceid=tcpid AND subchoice=0 ;
        IF isnull(xdesc,'') <> '' THEN
            SET rv=rv +xdesc+' '+tagchdesc
        END IF
    END CASE
;
    SET i=i+1 END FOR
END CASE
;
SET rv=rv+newline END FOR;
RETURN(rv)
END
}
```

From:

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

Permanent link:

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

Last update: **2026/08/07 19:24**

