



pears.TempTimesheetComplete



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"."TempTimesheetComplete" IS
{CREATE FUNCTION pears."TempTimesheetComplete"( IN @TempTimeSheetID CHAR(20)
)
RETURNS CHAR(250)
BEGIN
DECLARE "jobmasterid" CHAR(20);
DECLARE "placeid" CHAR(20);
DECLARE "persid" CHAR(20);
DECLARE "vacid" CHAR(20);
DECLARE @period INTEGER;
DECLARE "maxdate" DATE;
DECLARE "mindate" DATE;
DECLARE "TSNote" long VARCHAR;

DECLARE sminrate DOUBLE;
DECLARE paid DOUBLE;
DECLARE mperiod INTEGER;
DECLARE mpersid CHAR(20);
SELECT "placementid","period","personid","string"("serialnumber",' -
',"dateformat"("startdate",'dd/mm/yy'),' to
',"dateformat"("enddate",'dd/mm/yy')) INTO
"placeid",@period,"persid","TSNote" FROM "temptimesheet" WHERE
"temptimesheetid" = @TempTimeSheetID;
SELECT "vacancyid" INTO "vacid" FROM "placement" KEY JOIN "employment" WHERE
"placementid" = "placeid";
-- grade, speciality
UPDATE "awrjobmaster" SET "placementid" = "placeid" WHERE "personid" =
"persid" AND "vacancyid" = "vacid" AND "placementid" IS NULL;
-- grade, speciality
SELECT FIRST "awrjobmasterid" INTO "jobmasterid" FROM "awrjobmaster" WHERE
"placementid" = "placeID";
-- if there are duplicates ( should not be ) they need to be linked
IF(SELECT "sum"("unitspaid"*"payrate") FROM "temptimesheetline" WHERE
"temptimesheetid" = @temptimesheetid) > 0 THEN
IF "jobmasterid" IS NOT NULL THEN
```



```
-- use shift dates if avail
SELECT "min"("shiftdate"),"max"("shiftdate") INTO "mindate","maxdate" FROM
"tempshift" WHERE "temptimesheetid" = @temptimesheetid;
CALL
"AWRInsertWeekly"("isnull"("mindate","weekmonthenddate"(@period,'W')),"isnull"("maxdate","weekmonthenddate"(@period,'W'))','','','T','','jobmasterid',"TSNot
e",@TempTimeSheetID)
END IF
ELSE UPDATE "AWRWeeklyDetail" SET "disregard" = 1 WHERE "TempTimesheetID"
= (SELECT "OriginalTempTimesheetID" FROM "TempTimesheetCancellation" WHERE
"ContraTempTimesheetID" = @TempTimeSheetID)
END IF;

SET paid = 0;
SELECT MIN(temp-timesheetline.payrate), period, personid INTO paid,
mperiod,mpersid
FROM temptimesheet KEY JOIN temp-timesheetline KEY JOIN temp-payband AS
p
WHERE p.unit LIKE 'hour%' AND temptimesheet.temp-timesheetid =
@TempTimeSheetid
GROUP BY period, personid ;
IF (isnull(paid,0) > 0) AND EXISTS(SELECT * FROM pay_employee WHERE
personid = mpersid AND taxmethod = 1) THEN
SELECT GetMinWage(mpersid,mperiod) INTO sminrate;
IF (paid < sminrate) THEN
RETURN string('Temp not being paid the minimum wage of
&#163;',str(sminrate, LENGTH(sminrate)+3,2),'. Cancel the provisional
timesheet, change the rates against the placement and then complete the
timesheet. ')
END IF END IF;

RETURN NULL
END
}
```

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

Permanent link:
https://iqxusers.co.uk/iqxhelp/doku.php?id=database:procedures:pears_temp-timesheetcomplete

Last update: 2026/08/07 19:24

